COMPUTER-AIDED DESIGN OF MARINE REDUCTION GEARING

A Thesis Submitted
In Partial Fulfilment of the Requirements
for the Degree of
MASTER OF TECHNOLOGY

BY

LIEUT. D. M. S. NANDA

to the

DEPARTMENT OF MECHANICAL ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY KANPUR

AUGUST 1971

TO

MY PARENTS

CERTIFICATE

This is to certify that the thesis entitled "Computer-Aided Design of Marine Reduction Gearing" is a record of work carried out under my supervision and that it has not been submitted elsewhere for a degree.

J. Chakraborty
J. CHAKRABORTY
Assistant Professor
Department of Mechanical Engineering
Indian Institute of Technology
Kanpur

ACKNOWLEDGEMENTS

I am extremely indebted to my thesis adviser Dr.J. Chakraborty for his constant inspiration, guidance and for freely giving his valuable time for discussion throughout the work.

I am also thankful to Capt. V.M. Katdare, Indian Navy, for suggesting this problem and for his constant inspiration.

My thanks are also due to my colleagues, Mr. P.N. Singh, Mr. A. Chakraborty, Mr. M. Hariharan and Mr. B.S. Bhadoria for their valuable suggestions and criticisms from time to time.

My acknowledgements are also due to the staff of Computer Centre without whose cooperation this work would have been incomplete.

I am grateful to Mr. Uma Raman Pandey for his excellent and patient typing.

DARSHAN MOHAN SINGH NANDA

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

NOMENCLATURE

AK	Material constant
B	Face width
C_D	Centre distance
C_f	Surface condition factor
C_h	Hardness ratio factor
C_l	Life factor
C_m	Load distribution factor
C_{nf}	Face width coefficient
C_o	Overload factor
C_p	Elastic coefficient
C_r	Factor of safety
C_s	Size factor
C_t	Temperature factor
C_v	Dynamic factor
D_g	Gear operating diameter
D_p	Pinion operating diameter
$\bar{d}(\bar{X})$	Direction of move
E_g	Gear modulus of elasticity
E_p	Pinion modulus of elasticity
EPS	Accuracy of convergence criterion
FATS	Fatigue strength of material
$f(\bar{X})$	Objective function
G_f	Constraint
$\bar{G}(\bar{X})$	Gradient

HP	Horse Power
HPB	Horse power in bending
HPW	Horse power in wear
ITIR	Number of iterations
N	Number of design variables
NC	Number of constraints
PHIA(ϕ)	Pressure angle
PLV	Pitch line velocity
PMOD	Module
PND	Diameteral pitch
$P(\bar{X})$	Penalty function
$p(\bar{X})$	Penalty part of penalty function
r	Penalty parameter
RPM	R.P.M. of gear/pinion
SALB	Allowable stress
SC	Calculated stress number
SENS	Sensitivity factor
SHEAR	Allowable shear stress
SIA(ψ)	Helix angle
TRAN	Transmission ratio
U_g	Poisson's ratio for gear material
U_p	Poisson's ratio for pinion material
W_d	Dynamic load
$\bar{X}$	Design vector
$\bar{X}^*$	Optimum design vector

Y_p	Lewis form factor
Z_g	Number of teeth on gear
Z_p	Number of teeth on pinion.

Subscript 1 and 2 where used denote quantities relating to first and second set of gears respectively. Every variable which does not belong to the above group or has been differently used is defined at the place itself where it occurs.

SYNOPSIS

of the
Dissertation on
COMPUTER-AIDED DESIGN OF
MARINE REDUCTION GEARING
Submitted in Partial Fulfilment of
the Requirements for the degree
of
MASTER OF TECHNOLOGY IN MECHANICAL ENGINEERING
by

D.M.S. NANDA
Department of Mechanical Engineering
Indian Institute of Technology, Kanpur

August 1971

A compact and light gear box is one of the utmost consideration in the design of machinery space on board a Naval Ship. Compactness is achieved by a rational and systematic design of the gearing so that it occupies the minimum possible floor area. A mathematical model for the compactness of double reduction gearing set is formulated. Interior Penalty Function approach is used to arrive at the optimum size of the gears. The effectiveness of the proposed algorithm is proved by comparing the computer-aided design with the existing one.

CHAPTER 1

INTRODUCTION

Not long ago, weight was a critical factor mainly in Air Craft design. Today, weight reduction is the top design goal for a wide range of products starting from the 24' Marine gears to small wrist watches. Weight reduction usually means volume reduction, which in turn lowers cost of materials, handling and labour charges.

The designer of a Marine reduction gear box is mainly concerned with the above problem. The lighest gear box occupying the minimum deck space is considered the best design. The marine gears are required to take heavy and sudden loads, as in time of emergency, the ship is required to shoot to the maximum speed (say 25 knots) from stand still in an interval of 20-30 seconds thereby causing excessive impact on the gear teeth. The second requirement for naval gears is the quiet operation of the gear box to avoid detection of the ship's presence, due to the water borne noises, by the underwater operating submarines and other sophisticated gadgets employed by the enemy. And at the same time it is desirable to operate one's own ship quietly enough, to be able to hear water borne noises of the enemy's ships.

The bottom of a ship is generally of girder and steel plate construction and is therefore relatively flexible - a condition which may affect the design of the gear box. The position of the propeller shaft may influence the design by limiting the size of the bull wheel that can be accommodated between the shaft centre and the ship's bottom.

The first limitation in the design is met by providing flexible shafts and second by having the speed reduction in 2 stages, as the bull wheel in the 2 stage reduction is much smaller than the one in single reduction. Quietness in operation is achieved by making the gears with large number of teeth and cutting the teeth with extreme accuracy. The problem of marine gear design is complete if the above requirements are met and the components are designed for adequate power transmission.

1.1 Critical Review of Past Investigations

Ray C. Johnson[16], Staff Engineer, IBM, New York, has put forward a procedure for the optimum design of helical gears. But the author himself is not very sure of the reliability and accuracy of his method. His design procedure has certain inherent drawbacks. In his procedure of optimum design, he has incorporated all significant information in the equation form, and much of this information was originally available only in graphical or

tabular form, suitable for analysis or a cut-and-try type of design procedure. Secondly the author had put certain range restrictions on the parameters, constituting his design equations, which restrict the scope of the proposed procedure to be used for optimal design.

W.A. Tuplin[20] had suggested some curves for determining helix angles for minimum centre distance between the gear sets for various gear ratios and shaft angles. The method suffers from the defect that one has to decide the module and the gear ratios in the intermediate stages arbitrarily. The design so obtained can be far away from the optimal design as the designer would not know what specific combination of module and gear ratio will yield the optimal gear set.

E.J. Wellauer[9] had developed an analytical procedure using a number of strength rating formulae based on the wear rating formula developed by AGMA standards. The author claims that the design procedure obtained from his equations and charts result in an optimum design from the consideration of space occupied by the gear set. Though his method is workable for gears transmitting moderate horse powers but suffers from almost the same drawbacks as Johnson's optimal design procedure.

R.J. Wills(19) has suggested some new equations and charts which pick up the lightest weight spur gears. The method presented indicates gear sizes and ratios that will permit lightest possible design while still meeting the horse power and surface durability requirements. Though the equations and charts can be extended for designing helical gears but the method cannot incorporate the special requirements of marine gearing design and is suitable only for conventional gear design.

G.J. Huebner(18) has developed a computer oriented procedure for the design of gears. Balanced life concept of gear design is outlined in which gear and pinion are designed to fail simultaneously. The author has solved an example showing how the balanced life concept allows combination of minimum size and maximum capacity. But again the method suffers from the fact that special design requirements cannot be incorporated in this method. Moreover it cannot be used for multistage reduction gearing as the author does not suggest any criteria to choose the gear ratios in the intermediate stages of reduction.

1.2 Present Investigation - Its Scope

In the present dissertation a mathematical model of two stage marine reduction gear box has been developed. The algorithm takes into consideration the various features of marine gearing. All the component

gears have been designed for the maximum Horse Power. For this, a computer programme has been developed. Important correction data for the gear teeth has also been generated in the same programme.

In view of the fact that the compactness of the gear box leads to economy of space, machining time and weight, the minimum area occupied by the gear box has been chosen as the optimality criteria.

The optimal design problem is formulated as a mathematical programming problem. The optimisation algorithm chosen is the Fletcher and Powell's Variable Metric method. The method is iterative in nature and seeks the optimum from within the constraint set. To prove the effectiveness of the proposed algorithm, three example problems were taken up. First one refers to the reduction gearing for a Naval destroyer ship (40,000 H.P.), the second one is for a cargo ship (60,000 H.P.). In the third problem, the gears have been designed to transmit 75000 H.P., over a speed reduction ratio of 20.0.

CHAPTER II

MATHEMATICAL FORMULATION

In this Chapter, the design variables used in the mathematical modelling of the gear set, are introduced and the objective function is formulated. The design constraints are expressed in the form of inequalities and a penalty function to account for the constraints is generated. Also the solution scheme to solve the problem is discussed in brief.

2.1 Design Variables

The numerical quantities for which values are to be chosen in producing a design will be called 'Design Variables'. Other names have also been used, and, of course, the terminology used here is merely the author's preference. Perhaps the most descriptive of the alternative names are 'Construction parameters' or 'Construction Variables'.

The problem uses 11 variables. The design variables are the number of teeth on pinion in both the stages, the modules for both the stages, the transmission ratio in the first stage (the transmission ratio in the second stage being fixed from the consideration of overall transmission ratio); the helix angles for both the stages, the face width coefficients, the angle α (used for positioning the gear box centrally), and the error term which is the allowable manufacturing error for the gears.

It has been tactly decided to reduce the turbine speed to the propeller speed in two stages only. Experience in the field of marine gears confirms that it is advisable to reduce the speed in two stages only for a speed reduction ratio of the order of 15-30.

The pressure angle ϕ for the gears have been arbitrarily taken as 20 degrees. Here again it has been found that most of the heavy duty marine gears use 20.0° pressure angle because of the practical considerations.

2.2 Nomenclature of Important Components

The gears are named in single subscript. Z_{p1} and Z_{p2} stand for the number of teeth on pinion in the first and the second stage respectively. Similarly Z_{g1}, Z_{g2}, SIA1, SIA2, C_{nf1}, C_{nf2} stand for the number of teeth on gears, the helix angles, the face width coefficients for the first and the second stages respectively. PMOD1 and PMOD2 stand for the modules in the same way.

2.3 Design Space

If the number of independent variables in a particular configuration of a gear box is n, then the n-dimensional Eucledian space is known as the design space. Every point in the space represents a design even if it is absurd (as negative number of teeth or a gear digging into the other shaft). The movement from point to point in this space is represented by a vector

$$X = \left[X_1, X_2, X_3, \ldots\ldots, X_n\right]^T \qquad (2.1)$$

2.4 Constraint Surface

The set or locus of values of $\overline{X}$ which satisfy the equation $g_i(\overline{X}) = 0$ form a surface in the design space. This is not a 2-dimensional subspace, but an n-1 dimensional one where n is the number of design variables. It is a surface in the sense that it cuts the space into two regions, one where $g > 0$ (the acceptable region) and the other where $g < 0$ (the unacceptable region). Hence the surface that separates these two regions is called a constraint surface.

The portions of the respective constraint surfaces which bound the feasible region form a patch work of the constraints called the composite constraint surface. Points within this region (i.e. where $g_i(\overline{X}) > 0$), are called free points or unconstrained designs and points on the surface (i.e. designs which are such that at least one $g_i(\overline{X})=0$) are called bound points or constrained designs. It is possible for the acceptable regions or the feasible region to be composed of two or more disjoint subregions.

The sub-space where two or more $g_i(\overline{X})=0$ will be referred to as an intersection. The dimension of an intersection is n-r where r distinct constraints intersect.

2.5 Objective Function

Of all designs which are feasible some will be more desirable or better than others. If this is true then there must be some quality which the better designs have

more of, than do the less desirables ones have. If this quality can be expressed as a computable function of the design variables; we can consider optimizing to obtain a "Best Design". The function, with respect to which the design is optimized, is called the objective function. We will designate the objective function as $F(\overline{X})$. The best feasible design is one for which

$$g_i(\overline{X}) \geqslant 0 \quad , \quad i = 1,2,3,\ldots\ldots N_c$$

and

$$F(\overline{X}) \text{ is optimum.}$$

The selection of an objective function for an engineering design problem can be one of the most important decisions in the whole optimum process. In some situations, an obvious objective function exists because the need which the design is to fulfil is better served by some designs than others. In some design situations, there may appear to be two or more quantities which should be objective functions. Such situations are usually handled in one of the three ways :

(1) Composite objective function is formulated

(2) A limit is set for one of the functions and it is used as a constraint

(3) A trade off study is done.

In the present work the area occupied by the triangle formed by joining the centres of the three shafts is chosen as the optimality criterion (see fig. 2.1).

The following are the steps used to calculate the objective function :

$$TRAN2 = 20.0/TRAN\ 1$$

$$D_{p1} = Z_{p1}/(P_{nd1} \cdot \cos \psi_1)$$

$$D_{p2} = Z_{p2}/(P_{nd2} \cdot \cos \psi_2)$$

$$b_1 = C_{nf1} \cdot \pi \cdot \cos \psi_1 / P_{nd1}$$

$$b_2 = C_{nf2} \cdot \pi \cdot \cos \psi_2 / P_{nd2}$$

$$D_{g1} = D_{p1} \cdot TRAN\ 1$$

$$D_{g2} = D_{p2} \cdot TRAN\ 2$$

$$CD\ 1 = D_{p1} \cdot (TRAN\ 1 + 1.0)/2.0$$

$$CD\ 2 = D_{p2} \cdot (TRAN\ 2 + 1.0)/2.0$$

$$CD33 = CD1^2 + CD2^2 - 2CD1 \cdot CD2 \cdot \cos \alpha .$$

$$CD3 = \sqrt{CD33}$$

$$\cos \beta = \frac{CD1^2 + CD3^2 - CD2^2}{2CD1 \cdot CD3}$$

$$\beta = AR\cos(\cos \beta)$$

$$FO = \frac{CD3 \cdot CD1 \cdot \sin \beta}{2.0}$$

2.6 Normalised Constraints

The general form of expressions of design constraints is

$$g_1(\bar{X}) = 0.01 - F1(\bar{X}) \geqslant 0$$

$$g_2(\bar{X}) = F2(\bar{X}) - 10{,}000 \geqslant 0$$

where it is required that

$$F1\ (\bar{X}) \leqslant 0.01$$

$$F2\ (\bar{X}) \geqslant 10{,}000$$

The above is an example where the magnitude and sensitivity of the two constraints with respect to change in design variables are quite different. This results in diffuculty to arrive at the optimum point efficiently. The disparity in magnitudes and sensitivities of the design constraints can be avoided by normalising them, i.e., by forcing them to take values between 0 and 1. If the inequality constraint expresses the difference between two variables, then it should be normalised with respect to the variable of greater value. Even when the magnitudes of the constraints are controlled, their sensitivities may vary widely. Nevertheless, normalising always improves the handling of constraints.

2.7 Constraints

The design restrictions, the satisfaction of which distinguishes acceptable designs from the unacceptable designs, will be called collectively constraints. There are two useful categories of constraints in engineering problems. This grouping is not necessarily definitive, as it may not always be easy to classify constraints along these lines.

A constraint which restricts the range of design variables for reasons other than the direct consideration of performance of the design will be called a side constraint. Constraints which derive from those performance or behaviour requirements which are explicitly considered are called behaviour constraints.

Another type of constraint which arises in some engineering problems is that of the discrete valued design variables. In such cases the design variable is not to be selected from a continuous range of values but is permitted to take on only one of a discrete set of values. For example in the gear design problem one of such constraints is that the modules can only take some predetermined preferred values which do not form a continuous set.

The following is the list of side and behaviour constraints for the problem.

SIDE CONSTRAINTS : The face width coefficients for the marine gears vary from 4.0 to 12.0, the following constraints take care of this :

$$GX\ (5) = 1.00 - \frac{C_{nf1}}{12.0}$$

$$GX\ (6) = 1.00 - \frac{4.00}{C_{nf1}}$$

$$GX\ (7) = 1.00 - \frac{C_{nf2}}{12.0}$$

$$GX\ (8) = 1.00 - \frac{4.00}{C_{nf2}}$$

The minimum and the maximum values recommended for the helix angles are 15.0° and 35.0° respectively. Hence the constraints :

$$GX\ (9) = 1.00 - \frac{\psi_1}{35.0}$$

$$GX\ (10) = 1.00 - \frac{15.0}{\psi_1}$$

$$GX\ (11) = 1.00 - \frac{\psi_2}{35.0}$$

$$GX\ (12) = 1.00 - \frac{15.0}{\psi_2}$$

To achieve quietness in operation, a very high degree of accuracy in the manufacture of gears is desirable. The following constraints impose restrictions on the allowable error in manufacturing.

$$GX\ (21) = 1.0 - \frac{0.0005}{ERR}$$

$$GX\ (22) = 1.0 - \frac{ERR}{0.0012}$$

The most commonly used values for the length of the face varies from 1.5 P_{nd} to 6.00 P_{nd2}. Therefore the constraints :

$$GX\ (25) = 1.00 - \frac{b_1}{6.0 \cdot P_{nd1}}$$

$$GX\ (26) = 1.00 - \frac{1.5 \cdot P_{nd1}}{b_1}$$

$$GX\ (27) = 1.00 - \frac{b_2}{6.0 \cdot P_{nd2}}$$

$$GX\ (28) = 1.00 - \frac{1.5 \cdot P_{nd2}}{b_2}$$

To avoid too high a reduction taking place in the first stage alone, the following constraint has been imposed :

$$GX\ (35) = 1.0 - \frac{TRAN1}{7.00}$$

BEHAVIOUR CONSTRAINTS : To avoid interference of the gear teeth, the following constraints on the pinion teeth have been imposed :

$$GX(1) = 1.0 - \frac{13.7}{Z_{p1}}$$

$$GX(2) = 1.0 - \frac{13.7}{Z_{p2}}$$

The following constraints on the face contact ratio and the profile contact ratio have been imposed to assure positive control of the gear sets for all possible angular positions :

$$GX(3) = 1.0 - \frac{1}{FCR\ 1} + \frac{0.555\ PCR1}{FCR\ 1}$$

$$GX(4) = 1.0 - \frac{1}{FCR\ 2} + \frac{0.555\ PCR\ 2}{FCR\ 2}$$

The following constraints have been incorporated to make sure that the individual gears can transmit the designed H.P. in wear and bending.

$$GX(13) = 1.0 - \frac{WD1}{WW1}$$

$$GX(14) = 1.0 - \frac{WD2}{WW2}$$

$$GX(15) = 1.0 - \frac{HP}{HPW1}$$

$$GX(16) = 1.0 - \frac{HP}{HPW2}$$

$$GX(17) = 1.0 - \frac{HP}{HPB1}$$

The following constraints check the designed gears for adequate power transmission with the A.G.M.A. standards.

$$GX\ (19) = 1.0 - \frac{(S_{c1} \cdot C_{t1} \cdot C_{r1})}{SALB1 \ . \ C_{l1} \cdot C_{h1}}$$

$$GX\ (20) = 1.0 - \frac{(S_{c2} \cdot C_{t2} \cdot C_{r2})}{SALB2 \ . \ C_{\ell_2} \cdot C_{h2}}$$

GEOMETRY CONSTRAINTS : The following constraints prevent the gears from intersecting with the shafts of the adjoining gears :

$$GX\ (23) = 1.0 - \frac{0.6\ D_{g2}}{CD3}$$

$$GX\ (29) = 1.0 - \frac{2.0 \cdot DSH1}{D_{p1}}$$

$$GX\ (30) = 1.0 - \frac{2.0 \cdot DSH2}{D_{p2}}$$

$$GX\ (31) = 1.0 - \frac{2.0 \cdot DSH3}{D_{g2}}$$

$$GX\ (32) = 1.0 - \frac{D_{p2}}{2.0(CD1 - 15.0)}$$

The following three constraints have been incorporated so that the gearing set is laid out as centrally as possible :

$$GX\ (24) = 1.0 - \frac{ABS(0.5\ CD3 - CD1 \cdot Cos\beta)}{0.25\ CD3}$$

$$GX\ (33) = 1.0 - \frac{\alpha}{90.0}$$

$$GX\ (34) = 1.0 - \frac{30.0}{\alpha}$$

2.7-1 Positivity of Design Variables

Negative values for the number of teeth, transmission ratio and the modules of the gears are meaningless. To restrain the minimisation algorithm from going into this region of design space, positivity constraints are added.

$$G_i(\bar{X}) = \frac{X_i - (X_{min})_i}{(X_{max})_i - (X_{min})_i} \geqslant 0 \qquad (2.3)$$

$$i = 1,2,\ldots\ldots,N$$

where

N = number of independent variables

X_i = i^{th} design variable

$(X_{min})_i$ = minimum value of i^{th} design variable

$(X_{max})_i$ = maximum value of i^{th} design variable.

2.7-2 Gear Teeth

The gear has to have a minimum number of teeth to avoid involute interference. The following inequality if satisfied assures that there will be no undercutting of gear teeth.

$$\frac{4\ m}{\sin^2 \phi n} (N_{ng} + m) \leqslant 2\ N_{ng}\ N_{np} + N_{np}^{\ 2}$$

where

m = is a constant (Addendum Constant) of proportionality depending upon the standard gear tooth system. It equals 0.8 for 20° stub tooth.

ϕ_n= pressure angle referred to the normal plane

N_{np} = number of teeth in the formative gear set for the pinion

N_{ng} = number of teeth in the formative gear set for the gear

On simplification we get

$$N_{np}^{2} \not> \frac{4m(aN_{np} + m)}{(2a+1)\sin^2\phi_n}$$

where

a = Gear Ratio

Assuming a to be any high value say infinity we get

$$N_{np} \geqslant \frac{2(0.8)}{(\sin 20°)^2} = 13.7$$

$$\cong 14.0$$

This constraint is applied to all the gears in the system and is expressed as

$$GX\ (1) = 1.0 - \frac{13.7}{Z_{p1}}$$

$$GX\ (2) = 1.0 - \frac{13.7}{Z_{p2}} \qquad (2.4)$$

2.7-3 The Contact Ratio

Contact ratio is an important consideration in kinematic studies of helical gears. There are two significant types of contact ratios used in the terminology of helical gearing.

PROFILE CONTACT RATIO : The profile contact ratio - PCR, is defined as the arc of action divided by the circular pitch, both being measured in diametral plane. The profile contact ratio is an index for the average number of teeth in contact in any fixed diametral plane. The PCR is a function of the pitch helix angle Ψ, the number of pinion teeth N_p; the gear ratio a, the normal pressure angle ϕ_n and the addendum. Expressed mathematically PCR is given by $PCR = 0.555 \cdot a^{0.032} \cdot N_p^{0.104} \cdot \cos\Psi^{1.504}$.

FACE CONTACT RATIO : It is defined as the ratio of helical advance of a tooth element across the active face width to the circular pitch, both being measured in diametral plane. Expressed mathematically, face contact ratio FCR is defined by

$$FCR = \frac{b \tan\Psi}{P_c}$$

b = Active face width

P_c = Circular pitch

In order to assure positive control of the gear set for all possible angular positions, a definite

relationship between the contact ratios must be satisfied. From a kinematic consideration of helical gear sets, the stipulation to assure positive control of the gear set for all possible angular positions is

$$b \tan \geqslant P_c - \text{arc of action}$$

$$\frac{b \tan}{P_c} \geqslant 1 - \frac{\text{arc of action}}{P_c}$$

or $$FCR \geqslant 1 - PCR$$

Preferably for good control and smooth action,

$$FCR \gg 1 - PCR$$

It is often entirely satisfactory to have a profile contact ratio PCR less than unity provided FCR is large enough to satisfy the above equation.

2.7-4 Wear Strength

The load carrying capacity of helical gears is normally limited by pitting resistance of the tooth surface (wear). This resistance is commonly called the 'durability capacity.'

Pitting - Pitting is defined as surface fatigue failure of the material caused by repeated surface or sub-surface stresses that exceed the endurance limit of the material. It is characterised by the removal of metal and the formation of cavities. Pitting can be of two types :

Initial pitting - This may occur at the beginning of operation and continue only until the over stressed localized high areas of the surface have been reduced, thus

providing sufficient contact area to carry the load without further deterioration. Such pitting is not serious, because it is corrective and nonprogressive.

Destructive Pitting - This usually starts below the pitch line. The size and number of pits increase until smoothness of operation is impaired. The remaining surface fails in a similar manner, and finally the tooth shape is destroyed. The pits constitute stress raisers which can lead to failure by fatigue breakage.

Corrective and nonprogressive initial pitting is not considered serious. The AGMA formulae used here in the design are based on destructive pitting failures.

FAILURE - Pitting failure can be caused by any one or a combination of surface and subsurface shear, tensile and compressive stresses, depending on the specific distribution of inclusions, microstresses or other localized conditions.

Since the calculated and allowable stresses have a linear relation, the exact stress and location need not be determined. Instead the basic equation for contact stress is expressed as a 'contact stress number', S_c Its numerical value is equal to the surface compressive stress and is $3\frac{1}{3}$ times the maximum subsurface shear stress. The contact stress number is

$$S_c = C_p \sqrt{\frac{W_t\, C_O}{C_v} \left(\frac{C_s}{dF}\right) \left(\frac{C_m \cdot C_f}{I}\right)}$$

The relation between calculated and allowable contact stress number is

$$S_c \leqslant SALB \cdot \left(\frac{C_L \cdot C_H}{C_T \cdot C_R}\right)$$

The power capacity of gear set on wear basis is

$$HPW = \frac{Z_p \cdot F}{126{,}000} \left(\frac{IC_v}{C_s\, C_m\, C_f\, C_O}\right) \cdot \left[\left(\frac{SALB \cdot d}{C_p}\right)\left(\frac{C_L C_H}{C_T C_R}\right)\right]^2 \tag{2.5}$$

For a safe design, the constraint on HPW is

$HPW \geqslant HP$ for all the gear sets.

The notations used in the above expression had already been mentioned in the 'Nomenclature'.

A brief account of the various factors used in arriving at the expression of HPW is given here :

Overload Factor C_O : It evaluates the smoothness, roughness, and peak loads developed by the driving and the driven machines. The overload factors used have been taken from the tables given in the ASME transactions.

Dynamic Load Factor C_v : It depends upon the accuracy of tooth spacing and profile, pitch line velocity, inertia and stiffness of rotating elements, transmitted loads, tooth and blank stiffness, and properties of the lubricant. Commonly used dynamic factors are shown in Figure 2.2.

Size Factor C_s: It reflects the effect of dimensions and uniformity of material properties. For normal gears using good commercial material, as in this case, C_s equals 1. For practical design purposes, the necessary adjustments for size are usually included with the load distribution factor.

Load Distribution Factor : It measures the effects of nonuniform distribution of load across the face width. It is the ratio of the maximum to the average load intensity. Load distribution depends on -

1. Cutting errors
2. Internal Bearing clearance
3. Run Out
4. Tooth stiffness
5. Blank Stiffness and Shaft Stiffness
6. Bearing Deflections

The load distribution factor has also been taken from ASME transactions.

Geometry Factor : Includes the geometric terms developed in the formulation of relative radii of curvature, as well as the load sharing ratio between the oblique contact lines. The AGMA durability standards recommend

$$I = 0.225 \left(\frac{m_G}{m_G+1}\right)$$

where

m_G - gear ratio.

Surface Condition Factor CF : Depends on profile surface finish, residual stress, and plasticity effects. The surface finish is changed by operation under load, and usually stabilises at a definite value regardless of whether the initial finish is finer or rougher. Gears generated with reasonable care, as well as shaved can be rated with $C_f = 1.0 - 1.20$.

Life Factor C_L: Adjusts for the required number of tooth contact cycles. Usually the marine gears are designed for 10^8 cycles/day and the life factor used is 1.10.

Hardness Ratio Factor : Measures the hardness difference between the two meshing gears. In the present case, as the material of gear and pinion is the same and they have been subjected to the same degree of hardness, $C_h = 1$ is recommended for use.

Factor of Safety C_R: Enables the designer either to attain high reliability or if desired or necessary to assume a calculated risk of failure. Factors which must be considered in selecting a safety factor are : Effects of pitting on noise generation and uniform motion transmission and the possibility of stress concentrations initiating a tooth fracture.

The factor of safety of 1.25 has been used to take a higher reliability against failure.

Elastic Coefficient CP : Depends on the moduli of elasticity and Poisson's ratio of the pinion and gear material. It is given by the expression

$$C_p = \text{SQRT}\ (1/(\pi \cdot (\frac{1-\mu P^2}{E_p} + \frac{1-\mu G^2}{EG})))$$

2.7-5 Bending Strength

Loads that helical gears can transmit are normally limited by the pitting resistance of the tooth profile surface. Gear strength (bending considerations) becomes critical when materials of high hardness are used and for applications subject to high over loads. Though the gears will usually fail in wear, it is always advisable to check them from their strength point of view. The bending strength for a gear set is given as

$$HPB = \left[\frac{(RPM)^{.787}\ a^{.040}}{74,000}\right] \left[\frac{FATS^{1.26}}{C_3^{0.067}}\right] \left[\frac{D_p^{3.224}\ (C_{nF})^{1.192}}{(\cos\psi)^{2.004}} \cdot \frac{Y_n^{1.26}}{N_{np}^{2.450}}\right] \qquad (2.6)$$

where the symbols used have already been defined in the 'Nomenclature'.

For a 20° pressure angle

$$Y_N = 0.550 - 1.41/(N_n - 1.89)^{.756}$$

$$P(\bar{X}) = f(\bar{X}) + r \sum_{i=1}^{N_c} \frac{1}{g_i(\bar{X})}$$

At any constraint surface i, $g_i(\bar{X})$ reaches zero thereby shooting up the values of total function $P(\bar{X})$. The function $P(\bar{X})$ is given as high value as 10^{30} in case any one of the constraints is violated. This helps in identifying the situation that one or more constraints are violated. In the algorithm, the solution then starts from the previous feasible design.

2.9 Solution Scheme

The object is to minimise $f(\bar{X})$ subject to N_c constraints. The constraint list is far from being exhaustive. Even with these limited constraints, the solution becomes a formidable task because :

(a) The number of teeth on each gear must be an integer i.e., it should be 44 or 45 but not 44.67.

(b) The availability of cutters restrict the usage of standard modules. This is a situation where some of the variables can only take predetermined values.

These conditions make the formulation that of a nonlinear problem with integer and real variables. Some solutions for pure linear integer programming problems have been obtained but the nonlinear part is still unexposed.

The only feasible remedy for this impasse seems in floating the variables. After reaching the minima by

any of the existing techniques the variables are given their next higher integer values. But this method is also not free of faults. The final variables must satisfy all the constraints. Depending on whether the module value is made higher or lower to achieve the next standard value, the number of teeth of corresponding gears should be lowered or raised respectively. The output speed at the propeller must not differ by more than 2% from the required speed. Since the rounding-off of the teeth yields a change in the transmission ratio, there is a consequent change in the output speed. But these may not be as difficult as they appear to be. Because of the finite number of design variables, the simplest method would be to check every possible combination by fixing the range for each variable round the mean value found and always checking all the constraints. That which leads to minimum area amongst the acceptable combinations is the best design.

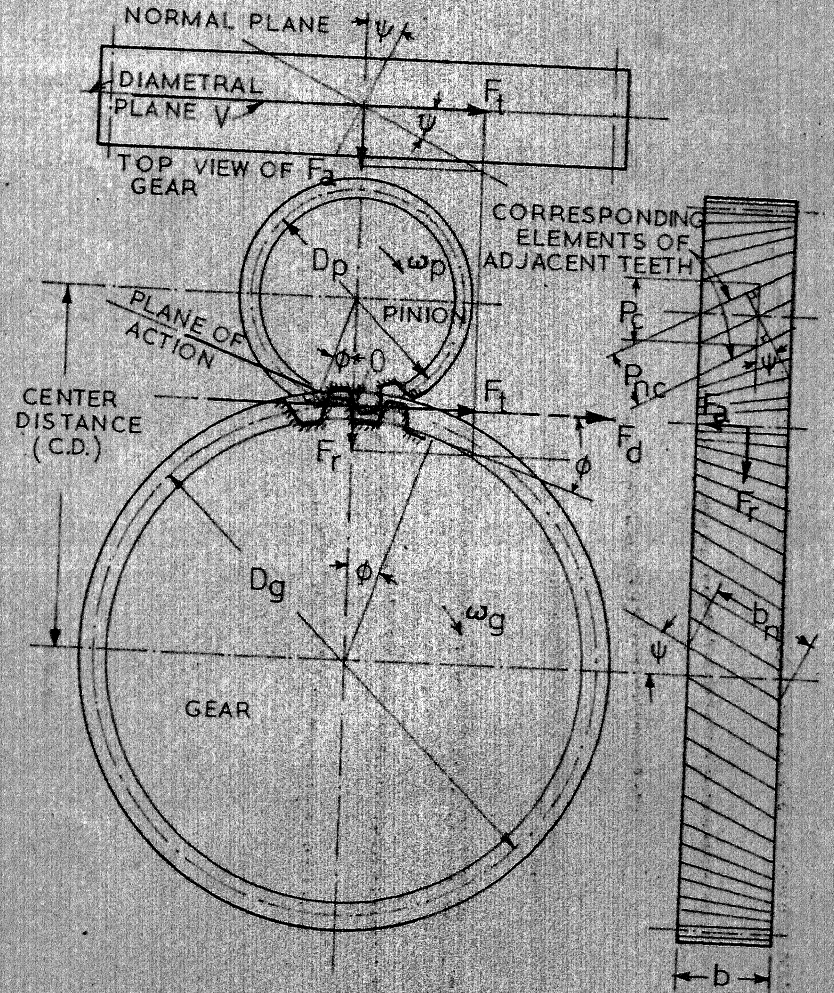


FIG. 1.1 TYPICAL HELICAL GEARSET SHOWING SIGNIFICANT FORCES PITCH LINE VELOCITY & SOME OF THE SIGNIFICANT GEOMETRY.

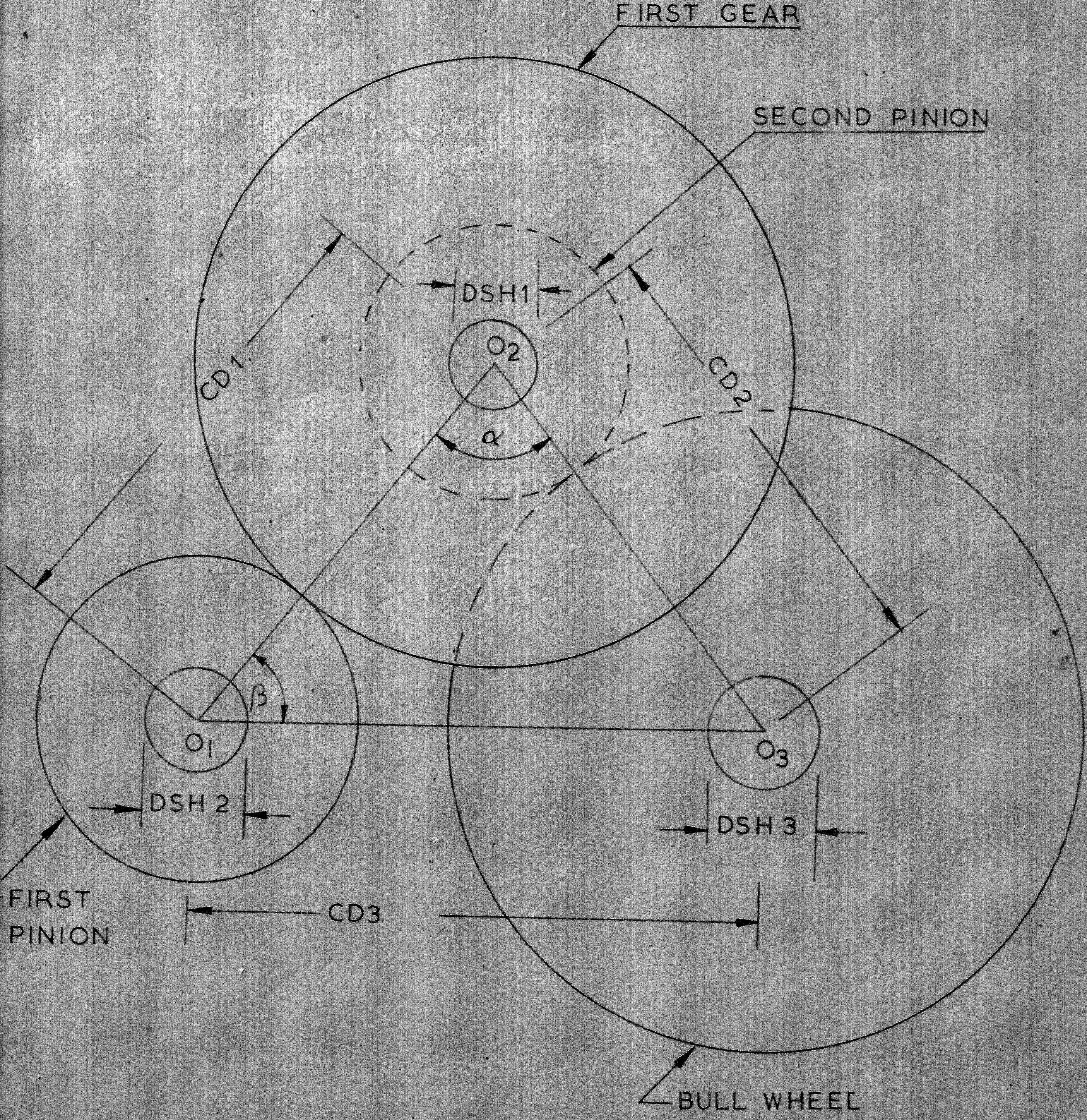
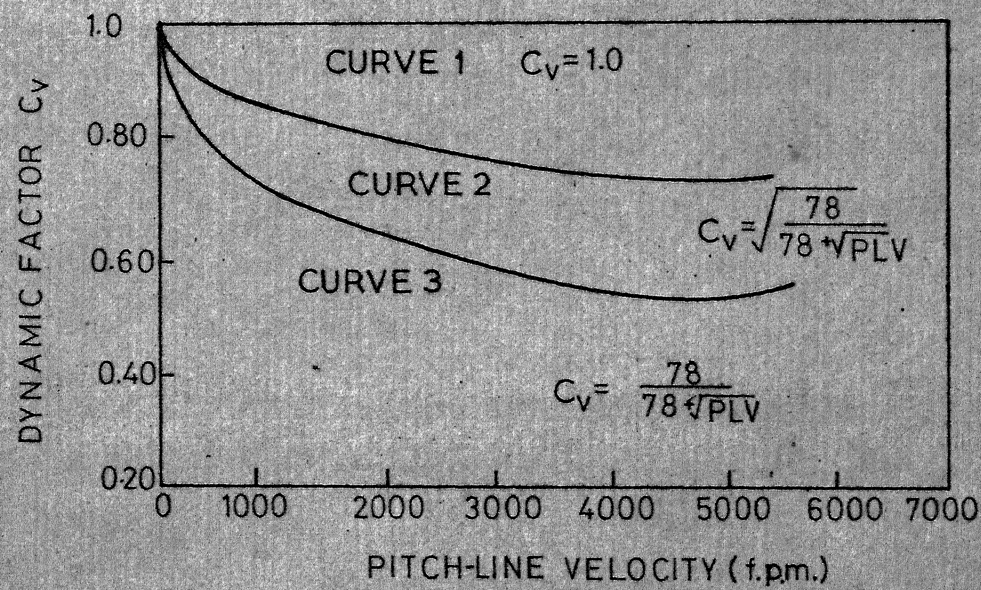


FIG. 2·1 LAYOUT OF DOUBLE REDUCTION GEARING



CURVE 1 – FOR PRECISION GEARS WITH LIGHT LOAD

CURVE 2 – LARGE PRECISION GAARS WITH LIGHT LOAD

CURVE 3 – GENERAL INDUSTRIAL & MARINE APPLICATIONS

FIG.2.2 COMMONLY USED DYNAMIC FACTORS

CHAPTER III

SOLUTION ALGORITHM

The technique employed in the solution of the foregoing constrained minimisation problem is discussed in this Chapter. The constrained problem is first converted to an unconstrained minimisation problem and is then solved as a sequence of unconstrained problems.

3.1 Conditions for Minimum

Necessary and sufficient conditions for minimum of a function $F(\overline{X})$ of n variables is that there exists $\overline{X}^*$ for which

$$f(\overline{X}^*) \leqslant f\,(\overline{X}) \qquad (3.1)$$

for all $\overline{X}$. If the inequality holds in some neighbourhood of $\overline{X}^*$ then the function is said to have a local or relative minima at that point.

The necessary conditions for the relative minimum are :

$$\frac{\partial f}{\partial X_i} = 0 \qquad , \quad i = 1,2,\ldots\ldots n \qquad (3.2)$$

The set of nonlinear simultaneous equations, thus obtained, are not easy to solve. Further, even if the above set could be solved there would be no guarantee that the obtained solution is not a maximum or saddle point rather than a minimum. One way to overcome this

difficulty is to reduce the constrained minimisation problem to an unconstrained minimisation problem.

3.2 Conversion to Unconstrained Minimisation

A constrained optimisation problem can be cast as a mathematical programming problem in the form

$$\text{Min} \quad f(\bar{X})$$

Subject to :

$$g_j(\bar{X}) \geqslant 0, \qquad j = 1,2,\ldots\ldots N_c \tag{3.3}$$

where $\bar{X}$ is an n-dimensional vector of design variables X_i, $i = 1,2,\ldots\ldots n$ and $g_j(\bar{X})$ are the given constraints on the design. The function $f(\bar{X})$ is called the objective function and its choice is governed by the nature of the problem.

The constrained minimisation problem is converted to an unconstrained minimisation problem as follows :

$$P(\bar{X},r) = f(\bar{X}) + r \sum_{i=1}^{N_c} \frac{1}{g_i(\bar{X})} \tag{3.4}$$

where $P(\bar{X},r)$ is the penalty function and r is an arbitrary penalty parameter which in the limit goes to zero.

The minimisation proceeds over a strictly monotanically decreasing sequence of r-values from an initial design vector $\bar{X}_0$ essentially, inside the boundary of the constraints and for a positive value of r, say r_1 (This is what is called 'Interior Penalty Function Approach). The minimum of $P(\bar{X},r_1)$ must lie inside the

constraint set because at the boundary some of the $g_i(\overline{X})$ tends to zero and consequently $P(\overline{X})$ tends to infinity. The minimum of $P(\overline{X},r_1)$ depends upon r_1, the starting value of r, and can be written as $\overline{X}(r_1)$. By reducing r_1 in the next iteration, the summation term which penalises for the closeness to the constraint boundaries is reduced and hence in minimising $P(\overline{X},r)$ more emphasis is placed on $F(\overline{X})$. The method has special advantages of not following the zig-zag pattern of minimisation (fig.3.1) as is observed with other nonlinear programming algorithms. This is because the minimisation starts from within and does not follow the boundary.

Fiacco and McCormick[6] have shown if

(1) the interior of the constraint set is non-empty,

(2) the functions $f(\overline{X})$ and $g_i(\overline{X})$, $i=1,2,\dots N_c$ are twice continuously differentiable,

(3) the set of points in the constraint set for which $f(\overline{X}) \leq V_o$ is bounded for every finite V_o and

(4) the function $f(\overline{X})$ is bounded below for $\overline{X}$ in the constraint set, then, the optimal solution to the unconstrained problem approaches a local minimum of the constrained problem as the value of r approaches zero. If, in addition,

(5) $f(\overline{X})$ and $-g_i(\overline{X})$, $i = 1,2,\dots N_c$ are convex functions, and,

(6) $P(\overline{X},r)$ is strictly convex in the interior of the

constraint set for every $r > 0$,
then the optimum solution to the unconstrained minimisation problem approaches the absolute minimum of the constrained problem as r approaches zero.

3.3 Choice of Initial r

To start the algorithm the first decision is to be taken about the value of r and the factor C, the factor by which the values of r shall be decreased. The algorithm, as such, imposes no restrictions on the values of r and C except that $r > 0$, $C > 1$. If r is large, the function is easy to minimize, but the minimum may lie far from the desired solution to the original constrained minimisation problem. On the other hand if r is small, the function will be hard to minimize as it then requires the initial solution to be very close to the actual minima.

Two methods to choose the initial value of r have been suggested in the literature.

(1) $$r_1 = \frac{\nabla f(\bar{X}_0)^T \cdot \nabla p(\bar{X}_0)}{|\nabla p(\bar{X}_0)|^2} \qquad (3.5)$$

where

$$\nabla p(\bar{X}_0) = \frac{\partial}{\partial x_i} \sum_{i=1}^{N_c} \frac{1.0}{g_i(\bar{X}_0)}$$

and represents the gradient of the function.

<u>Case 1</u> : $r_1 < 0$

If $r_1 < 0$, minimisation of $f(\overline{X}_0)$ alone, without considering the penalty term, is carried out. At every new point the r value is checked by equation 3.5. If the value is positive, unconstrained minimisation with obtained value of r is carried out.

<u>Case II</u> : $r_1 = 0$

This means that the unconstrained minimum has been reached and

$$\overline{X}_0 = \overline{X}^*$$

<u>Case III</u> : $r_1 > 0$

If $r_1 > 0$, this is taken as the starting value for usual minimisation.

(2) If the initial design is conservative(i.e., not near any constraints), one would like to pick the initial $r = r_0$ so that $F_{min}(r_0)$ would not increase drastically over the original design. In other words, r ought to be chosen small enough that in the neighbourhood of the initial design, the $r \sum \frac{1}{g_i(\overline{X})}$ terms do not completely dominate $P(\overline{X}, r_0)$. A rule[1] which might follow from this observation is that if $\overline{X}_0$ is a conservative design, pick r_0 so that $r_0 = r_1$ is given by

$$r_1 = \frac{f(\overline{X}_0)}{\sum_{i=1}^{N_c} \frac{1}{g_i(\overline{X})}} \qquad (3.6)$$

This has two weaknesses. Firstly, the starting point may be too close to some of the boundaries thereby making such selection of r useless, because the r value dictated by the above rule might be too small to allow the first minimisation to be carried out. In this case, a proper value of r_0 will likely be large enough that in minimizing $P(\overline{X},r)$, F will increase from its value at $\overline{X}_0$. While this is distressing, it probably cannot be helped with this form of penalty function without a good deal of complex logic. Furthermore, unless something really happens, very little is lost since r can be reduced quite quickly in this method by giving a reasonable value to C.

Another approach to this latter problem (initial solution being close to constraint surfaces) which seems appealing in some cases is to pick a relatively large value of r but to temporarily add a new constraint to the problem in the form of

$$g_{Nc+1} = F(\overline{X}) - F(\overline{X}_0) \geq 0$$

or to make it easier to get a starting point

$$g_{Nc+1} = F(\overline{X}) - (F(\overline{X}_0) + \epsilon) \geq 0$$

where ϵ is some small amount of increase which will theoretically be permitted in F on the first minimization. The penalty function for this revised problem is then

$$P(\overline{X},r) = F(\overline{X}) + r\left\{\sum_{i=1}^{N_c} \frac{1}{g_i(\overline{X})} + \frac{1}{F(X) + \left\{F(\overline{X}_0) + \epsilon\right\}}\right\}$$

The minimum for large values of r is approximately the point where the term in brackets is a minimum. As r is decreased, the fictitious constraint term can be removed or left in as desired since it will ultimately vanish.

3.4 Extrapolation For r

The vectors $\bar{X}(r_1)$, $\bar{X}(r_2)$ ..... $\bar{X}(r_K)$ obtained by minimising $P(\bar{X}, r)$ over decreasing sequence of r-values lie on a trajectory $\bar{X}(r)$, where $\bar{X}(0)$ is the desired solution. From this trajectory, it is possible to obtain, by extrapolation, estimates of final solution $\bar{X}(0)$ and the next minimum $\bar{X}(r_{K+1})$.

It has been shown that the trajectory approximates the polynomial of $r^{1/2}$ (3). This is because as r tends to zero, the function behaves as a linear function of $r^{1/2}$. The approximating function is defined as

$$\bar{X}(r)=a_0+a_1r^{1/2}+a_2r + \ldots\ldots a_{K-1}r^{\left(\frac{K-1}{2}\right)} \tag{3.8}$$

where a_i's are the undetermined coefficients.

For small values of r

$$\bar{X}(r) = \bar{X}(0) + a\sqrt{r} \tag{3.9}$$

Solving these for $\bar{X}(0)$

$$\bar{X}(0) = \frac{\sqrt{C}\ \bar{X}(r/C) - \bar{X}(r)}{\sqrt{C} - 1} \tag{3.10}$$

An estimate of minimum point for next r-value is obtained by assuming that

$$\bar{X}(r/C^2) = \bar{X}(0) + a\sqrt{r/C^2} \tag{3.11}$$

Solving equations (3.10) and (3.11) for $\bar{X}(r/C^2)$ we get,

$$\bar{X}(r/C^2) = \bar{X}(r/C) + \frac{1}{\sqrt{C}} (\bar{X}(r/C)) - \bar{X}(r) \tag{3.12}$$

This value of X can be used for the next minimisation, thereby substantially reducing the effort for minimisation of $P(\bar{X})$. However these extrapolated values have to be first checked so that they do not violate the constraints.

Estimates of $\bar{X}(0)$ become reasonably good after 3 or 4 trials, and towards the end offer more accurate estimation of values of $\bar{X}(0)$.

3.5 Method of Unconstrained Minimisation Davidon-Fletcher-Powell Variable Metric Method

This method completely avoids the need for evaluating second derivatives and performing matrix inversions and yet the sequence of iterates converges quadratically to the minimum point, $\bar{X}_m$.

The method is based on the properties of a quadratic function and is designed so that when applied to a quadratic, it minimises the function of n variables in 'n' iterations.

Central to the method is a symmetric positive definite matrix $[H_i]$ which is updated at each iteration. It collects in itself the information about the curvature of the function $P(\bar{X})$ and supplies the current

direction of move d_i by multiplying it with the current gradient vector G_i.

A typical iteration in the minimisation of $F(\overline{X})$ proceeds as follows :

(1) Given the starting point $\overline{X}_i$ and the gradient of $P(\overline{X})$ at $\overline{X}_i$, $d_i(\overline{X})$, the direction of $(i+1)^{th}$ minimisation of $P(\overline{X})$ is given by

$$\overline{d}_i = - \left[H_i\right] \cdot G_i \tag{3.13}$$

(2) Find σ_i^* so that $F(\overline{X}_i+\sigma_i^* d_i)$ is minimum along the line $\overline{d}_i$.

(3) Set $\overline{X}_{i+1} = \overline{X}_1 + \sigma_i^* d_i$.

(4) Calculate the new gradient vector

$$\overline{G}_{i+1} = \nabla P(\overline{X}_{i+1})$$

and set

$$\overline{Y}_i = \overline{G}_{i+1} - \overline{G}_i$$

(5) Calculate the new $[H]$ matrix, $[H_{i+1}]$ by

$$\left[H_{i+1}\right] = \left[H_i\right] + \sigma_1^* \frac{\overline{d}_i \, \overline{d}_i^T}{\overline{d}_i^T \, \overline{Y}_i} - \frac{\left[H_i\right] \overline{Y}_i \, \overline{Y}_i^T \left[H_i\right]}{\overline{Y}_i^T \left[H_i\right] \overline{Y}_i} \tag{3.14}$$

(6) Begin the next iteration from (1)

The basic algorithm is extremely powerful for a first order method, converging quadratically and possessing very good stability. By stability I mean that even in highly distarted and eccentric functions, it continues to progress and needs little of the sort of special attention as required by its parallel algorithm, the

conjugate gradient method. There is a plausible argument for this increase in stability in that with conjugate gradients, the entire history of the path is carried to d_{i+1} in the intelligence of $\beta_i\, d_i$, a single vector ($\beta_i = |G_{i+1}|^2 / |(G_i)|^2$). In this variable metric method, we carry the data in a full matrix which we carefully upgrade at each iteration. Another advantage of this method is that in conjugate gradients the carry over term $\beta_i d_i$ is only good if applied to $\nabla P(X_i)$ and produces non-sense if applied to gradient at some other point, on the other hand it can be shown that $\{H_i\}$, is a positive definite approximation to the matrix of second order partial derivatives and is applicable in the whole space.

There are certain drawbacks in the algorithm which must be kept in mind while applying this method. The positive definiteness of $\{H\}$ matrix is preserved in theory only if σ_i^* is the true minimum point (i.e. $G_{i+1}^T \cdot d_i = 0$) and furthermore, round off error will again dog our steps so that the process can get into trouble.

In applying the method, therefore, care must be exercised to ensure that the $\{H\}$ matrix is not updated with data arising from poor approximations to σ_i^*. There are a number of approaches to this problem : First the algorithm used for computing σ_i^* may be reapplied until $d_i^T \cdot G_{i+1}$ is sufficiently small, another alternative is simply to skip the update cycle {step 3 in the

algorithm} when $d_i^T G_i$ is too large. In other words if σ_i^* is not close enough to minimum along $\bar{d}_i$, set $H_{i+1}=H_i$ and $d_{i+1} = -H_{i+1} \cdot G_{i+1}$ and continue as before. As long as $P(\bar{X}_{i+1}) \leqslant P(\bar{X}_i)$, the method will continue to progress towards the minimum.

It is difficult to choose between these approaches, the first may require excessive computation to refine σ_i^* at points far from $\bar{X}_m$ while on the other hand, the second approach may pass up valuable opportunities to improve the {H} matrix. A reasonable compromise is to set a moderate criterion for $d_i^T \cdot G_{i+1}$, limit the number of refits for σ_i^* and then skip the update cycle if the criterion is not met on the second try (This logic has been used in the programme).

3.6 The Gradient

At every iteration the gradient has to be calculated in the minimisation algorithm. It can best be determined by writing down the exact derivative of $P(\bar{X})$ with respect to the design variables. Because of the large amount of work involved in calculating the required partial derivatives, it has been decided not to derive analytic expressions for the components of the gradient vector, but instead, to approximate the gradient by difference equations which involve only function evaluations.

The gradient of the function $P(\bar{X})$ with respect to the design variables is given by:

$$G(\bar{X}) = \nabla P(\bar{X}) = \left(\frac{\partial P}{\partial X_1}, \frac{\partial P}{\partial X_2}, \ldots \frac{\partial P}{\partial X_n}\right)^T \qquad (3.15)$$

It may also be written as

$$P(\bar{X}) = \left(\frac{\partial F}{\partial X_1} - r \sum_{i=1}^{N_c} \frac{1}{g_i^2} \frac{\partial g_i}{\partial X_1}\right), \ldots\ldots,$$

$$\left(\frac{\partial F}{\partial X_n} - r \sum_{i=1}^{N_c} \frac{1}{g_i^2} \frac{\partial g_i}{\partial X_n}\right) \qquad (3.16)$$

The variable metric method technique for minimisation is based upon the assumption that the function being minimised can be satisfactorily represented by a quadratic function in the vicinity of the function minimum. Therefore expanding the function $F(\bar{X})$ of one variable about point X_0 by Taylor series up to quadratic terms we have

$$F(X_1) = F(X_0) + \left(\frac{dF}{dX}\right)\Big|_{X_0} (X_1 - X_0) + \frac{1}{2} \frac{d^2F}{dX^2}\Big|_{X_0}(X_1 - X_0) + \ldots \qquad (3.17)$$

Neglecting the second order and higher order terms

$$\frac{dF}{dX}\Big|_{X_0} = \frac{F(X_1) - F(X_0)}{(X_1 - X_0)} \qquad (3.18)$$

Close to the minimum, the function $P(\bar{X})$ has very large curvature indicating significant values for the second and third derivatives which have been neglected in the forward difference scheme. Even if $g(\bar{X})$ and $f(\bar{X})$ are nonlinear, still they are quite well behaved as compared to $P(\bar{X})$ at any point inside the domain. Thus the gradient from the equation (3.16) should be used.

3.7 Linear Minimisation (Fibonacci Search)

The one-dimensional search procedure used in the programme is the method called the Fibonacci Search. Here we make use of the Fibonacci Numbers, which are defined by

$$F_{N+1} = F_N + F_{N-1} , \quad F_0 = F_1 = 1$$

The principle of the working of this method is as follows :

Assume that the minimum of F(x) along x lies between two points A and B. According to the accuracy ϵ_x of the result required, choose a value of N, such that

$$\frac{x_A - x_B}{F_N} < \epsilon_x$$

Choose two points 1 and 2 at distances α_1 and α_2 from A.

$$\alpha_1 = \left(\frac{x_A - x_B}{F_N}\right) \cdot F_{N-1} \qquad \alpha_2 = \left(\frac{x_A - x_B}{F_N}\right) \cdot F_{N-2}$$

Compute the function values at 1 and 2 as F_1 and F_2. Assuming that the function is unimodal, i.e. the point with smaller function value to be nearer the minimum compare F_1 and F_2. The portion outside the larger value is discarded, and the search is repeated within the rest of the zone, until the minimum is reached to the given accuracy (Fig.3.2).

3.8 Slope in $\bar{d}$ Direction:

In one dimensional minimisation of $P(\bar{X})$, slope in the direction d is required. This can be obtained in two ways :

1. By taking dot product of the vector $\bar{d}$ and $\bar{G}$ at the point.

$$\psi' = \bar{d}^T \cdot \bar{G}$$

2. By taking finite difference in $\bar{d}$ direction.

In the first method, to calculate the gradient, it is essential to evaluate the function value n times. Whereas, in the second method only one computation of the function is necessary for the forward difference scheme. Therefore the second method has been preferred here.

The forward difference scheme in the $\bar{d}$ direction is as follows :

$$\bar{X}_i^{+} = \bar{X}_i + \epsilon \cdot \bar{d}_i$$

where ϵ is a small step in $\bar{d}$ direction. The slope $\psi'(\sigma)$ in $\bar{d}$ direction is then

$$\psi' = \frac{d\psi(\sigma)}{d\epsilon} = \frac{df(\epsilon)}{d\epsilon} - r \sum_{i=1}^{N_c} \frac{dg_i}{d\epsilon} \frac{1}{g_i^2} \qquad (3.23)$$

where

$$\frac{df(\bar{X})}{d\epsilon} = \frac{f(\bar{X}^{+})-f(\bar{X})}{\epsilon}$$

$$\frac{dg_i}{d\epsilon} = \frac{g_i(\bar{X}^{+})-g_i(\bar{X})}{\epsilon}$$

3.9 Convergence Criteria

The following are the various convergence criteria used in the algorithm at various stages.

1. For one dimensional Minimisation

a) $$\left| \frac{\bar{d}^T \cdot \bar{G}}{\bar{d} \cdot \bar{G}} \right| < \epsilon_{S1} \qquad (3.24)$$

where ϵ_{S1} is the cosine of angle between $\bar{d}$ and $\bar{G}$.

b) The percent change in the function is less than specified tolerance

$$\left| \frac{f(X^+) - f(X)}{f(X)} \right| \leqslant \epsilon_{s2} \tag{3.25}$$

2. For minimisation for a particular r

a) When the approximation for the percent change in $P(\bar{X})$ in the next iteration has become less than a certain pre-assigned value.

$$\frac{\bar{G}[H]\bar{G}}{P(\bar{X})} \leqslant \epsilon_{m1} \tag{3.26}$$

b) Another criterion to stop the iterations for a particular r is when n/2 consecutive one-dimensional minimisations have yielded α^* (step in the $\bar{d}$ direction) less than one percent of the smallest value of the design variable.

3. For the Unconstrained Minimum

a) When the percent change in the objective function between minima corresponding to two consecutive r values is less than the given tolerance, that is

$$\left| \frac{f_{min}(\bar{X}, r-1) - f_{min}(\bar{X}, r)}{f_{min}(r)} \right| \leqslant \epsilon_{u2} \tag{3.27}$$

b) When the minimum percent change in any design variable corresponding to two consecutive minimum r values is less than the given tolerance.

$$\left| \frac{X_i(r/C) - X_i(r)}{X_i(r)} \right| \leq \in_m \tag{3.28}$$

c) When the percent change in the extrapolated function value f(0) is less than the given tolerance.

$$\left| \frac{f(0) - f(r)}{f(r)} \right| \leq \in_r \tag{3.29}$$

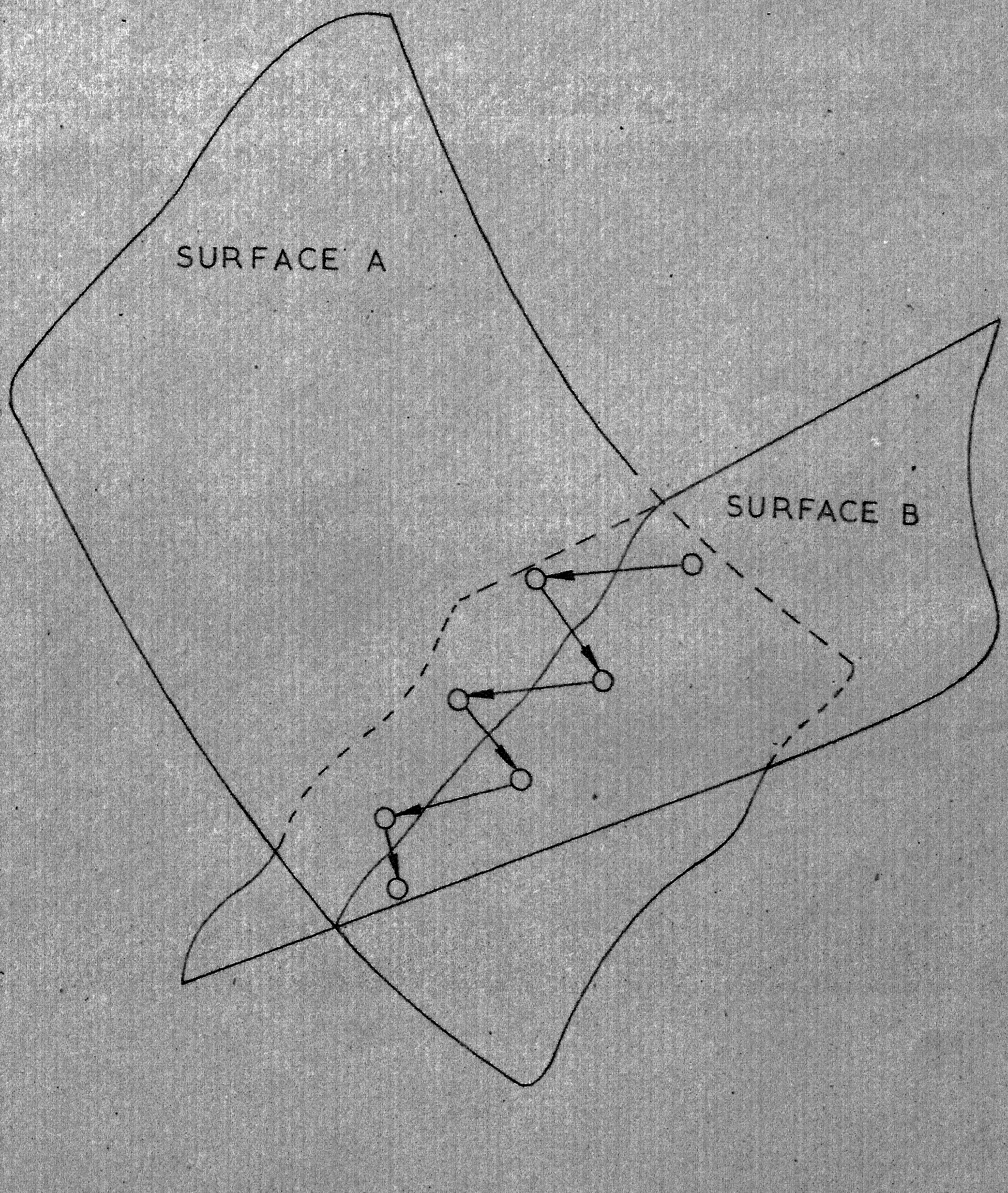


FIG. 3·1 ZIGZAGGING

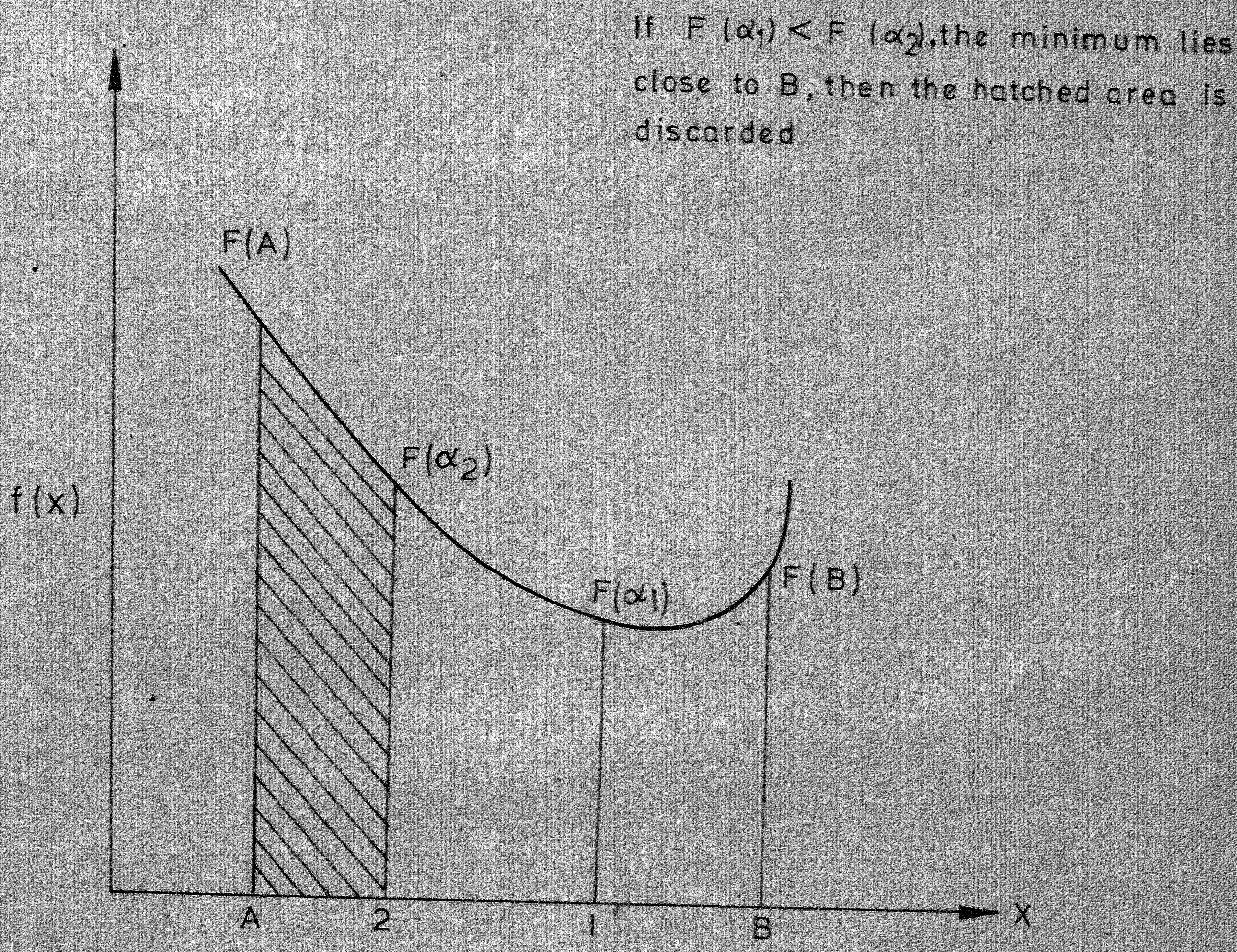


FIG 3.2 ONE DIMENSIONAL MINIMISATION

CHAPTER IV

RESULTS AND DISCUSSIONS

Three illustrative examples have been solved to show the applicability of the unconstrained minimisation algorithm to solve the marine gearing problem. First example is the main reduction gearing of a Naval destroyer. Results obtained therefrom are tabulated. In the second example the gear box for a cargo ship - JAL DEEP, having 60,000 H.P. has been designed and the results compared with the existing design. In the third example the gears have been designed to transmit 75,000 H.P., but the results could not be compared as the existing results for this were not available.

In all the previous three examples, the speed reduction had been from 3000.0 R.P.M. to 150.0 R.P.M., the gear ratio being 20.0. The criterion for 'best' design in all the discussed examples is the minimum deck area occupied. Constants which are common to all the three examples have been tabulated in Table 4.1. Table 4.2 shows different initial starting vectors for the discussed examples.

4.1 Example 1

This is the main drive for a Naval destroyer having 40,000 H.P. at its propeller. The drive has two

reduction stages, each stage employing double helical gearing. Input information for this problem is given in Table 4.1 and 4.2. Table 4.3 compares the existing and optimal results. Table 4.6 shows various optimum solutions.

4.2 Example 2

This is the reduction gearing for JAL DEEP, a cargo ship with 60,000 H.P. at its single screw. The speed reduction is carried out in two stages so as to facilitate the accommodation of bull wheel in the space between the centre of the shaft and the bottom of the ship. Double helical gears have been used. Table 4.4 compares the existing and the optimum results. Table 4.7 shows the variation in the optimum solution with the variation of the starting value of 'R' and different starting vectors.

4.3 Example 3

This is for the drive of a single screw ship with 75,000 H.P. The speed reduction had been carried out in two stages using herringbone gears. Table 4.8 shows the final optimum results obtained by the variations of 'R' and starting vectors.

4.4 Discussions

The results from the first example show about 23.4% reduction in area from the existing design. The results of second example also show a reduction of 54.3%

in area from the existing design. The rounded numbers of teeth have adjusted themselves in such a fashion that out of two mating gears, the numbers of their teeth have no simple common factor, which is desirable from the wear considerations. This added advantage shall be obtained almost always because the transmission ratios in both the reduction stages do not arrange themselves in simple fractions. The same objective could also be achieved by adding a hunting tooth on the gear, in case there is any common factor between the number of teeth on gear and pinion.

The objective function in the first example decreases only by 23.4% whereas in the second example, the reduction is as high as 54.3%. This is so because the first example was for a Naval ship wherein the utmost attention had been paid to save as much space as possible, whereas, the second example is for a cargo ship where different criterion, other than space, must have dominated in the design philosphy. The optimum results for example 3 could not be compared due to unavailability of the existing results.

The percentage improvement in the objective function in the discussed examples, may be attributed to one or more of the following points.

1. The method of existing gear design may be different from the one used here.

2. In the present algorithm, the computer facility has been fully exploited to find out the best combination of the independent design parameters to achieve the optimum design. In conventional design procedures, this is not possible.

3. Better material with 550 BHN has been used in the present design compared to conventional gear materials with BHN of 400-500.

The optimum search is possible here because the algorithm seeks the minima (at least local) in the feasible design space.

The author does not claim that the global minima had been reached in the examples discussed. This is because firstly not many starting vectors were chosen (only 3 starting vectors for each example) due to shortage of computer time and even for one starting point the iterations were stopped when no significant changes occured in the objective function. Secondly even in some of the examples with a particular starting vector, the convergence was rather slow as compared to other starting vectors for the same example. This may be due to the inefficiency of the linear minimisation algorithm which prevented the process from making satisfactory progress towards the optimum solution. It is found that when the starting point of the linear minimisation lies close to a constraint boundary, so that even a very small

step away from this point causes a constraint to be violated, the linear minimisation procedure comes up with a poor approximation to the minimum of the function along that line.

Another problem which was encountered in using the present method is the choice of an initial value of penalty parameter 'R'. In Chapter 3, two methods of selecting the initial value of 'R' have been discussed, both of which depend upon some knowledge of the characteristics of the gradient vector. In using the finite difference formula to compute the gradient at each step, these necessary facts are not available. It would be better if a more rigorous method for choosing the starting value of 'R' could be found.

TABLE 4.1

VALUES OF CONSTANTS COMMON TO ALL EXAMPLES

SYMBOL	IDENTIFIER	MAGNITUDE
ϕ	PHIA	20.0°
	RPM 1	3000.0
	RPM 3	150.0
E_g	EG	29000000.0 lb/in^2
	AK	861.00
C_3	C3	1.67
U_g	UG	0.20
U_p	UP	0.20
	SALB	225,000.0 lbs/in^2
	FATS	150,000.0 lbs/in^2
	SHEAR	850,000 lbs/in^2
	ITIR	200
	TRAN	20.0
	EPS1	0.0005
	EPS2	0.0005
	EPS3	0.0005
	EPS4	0.0005
C_o	CO	1.50
C_r	CR	1.25
C_l	CL	1.10
C_s	CS	1.00
C_h	CH	1.00
C_t	CT	1.00
C_f	CF	1.00
	SENS	0.50
	PMOD	1.0, 1.125, 1.375, 1.50, 1.75, 2.0, 2.25, 2.50, 2.75, 3.00, 3.25, 3.50, 3.75, 4.00, 4.5, 5.0, 5.5, 6.0, 6.5, 7.0, 7.5, 8.0, 8.5, 8.0, 9.5, 10.0, 11.0, 12.0, 13.0, 14.0, 15.0, 16.0

Column of symbol is kept blank where the symbol and the identifier are the same.

TABLE 4.2

DIFFERENT INITIAL DESIGN VECTORS FOR EXAMPLES 1, 2 AND 3

Representation $\bar{X}$ = {ZP1, ZP2, PND1, PND2, SIA1, SIA2, TRAN1, ALPHA, CNF1, CNF2, ERR}

EXAMPLE 1 :

	$\bar{X}_1$	$\bar{X}_2$	$\bar{X}_3$
ZP1	50.0	45.0	41.62
ZP2	80.0	80.0	86.65
PND1	2.40	2.45	2.314
PND2	2.40	2.45	2.312
SIA1	25.0	24.0	22.81
SIA2	25.0	24.0	22.86
TRAN1	5.50	5.50	5.87
ALPHA	55.0	52.00	52.27
CNF1	8.0	8.00	10.90
CNF2	9.50	8.50	11.55
ERR	0.00065	0.00070	0.00055

EXAMPLE 2:

	$\bar{X}_1$	$\bar{X}_2$	$\bar{X}_3$
ZP1	100.0	75.0	42.0
ZP2	130.0	100.0	86.0
PND1	2.0	2.20	2.20
PND2	2.0	2.20	2.315
SIA1	28.0	26.00	22.80
SIA2	28.0	26.00	22.80
TRAN1	5.0	5.00	5.89
ALPHA	70.0	65.00	52.50
CNF1	8.0	8.00	10.00
CNF2	8.0	8.00	11.55
ERR	0.00085	0.00080	0.00055

EXAMPLE 3 :

	$\bar{X}_1$	$\bar{X}_2$	$\bar{X}_3$
ZP1	150.0	41.62	60.0
ZP2	150.0	86.60	120.0
PND1	2.0	2.314	2.50
PND2	2.0	2.312	2.45
SIA1	30.0	22.81	25.00
SIA2	30.0	22.86	25.00
TRAN1	5.0	5.87	6.00
ALPHA	65.0	52.27	55.00
CNF1	8.0	10.90	10.50
CNF2	8.0	11.56	11.00
ERR	0.00085	0.00055	0.00055

N.B. - The subscripts 1,2,3 stand for starting vector 1,2 and 3.

TABLE 4.3

COMPARISON OF OPTIMUM DESIGN TO THE EXISTING DESIGN FOR EXAMPLE 1

S.No.	DESCRIPTION	EXISTING DESIGN Ist Stage	EXISTING DESIGN 2nd Stage	OPTIMUM DESIGN Ist Stage	OPTIMUM DESIGN 2nd State
1.	Number of teeth on pinion	58	115	44	80
	Number of teeth on gear	290	460	259	281
2.	Module	7.8*	7.8*	10.0	10.0
3.	PCD of pinion in m.m.	452.4	897.0	440.0	800.0
	PCD of gear in m.m.	2262.0	3588.0	2590.0	2810.0
4.	Helix angle in degrees	27.93°	27.93°	22.90	22.90
5.	Transmission ratio	5.0	4.0	5.78	3.46

*Obtained by converting the circular pitch into equivalent module.

N.B. - The Optimum Design refers to Initial Design Vector-2

TABLE 4.4

COMPARISON OF OPTIMUM DESIGN TO THE EXISTING DESIGN FOR EXAMPLE-2

S.No.	DESCRIPTION	EXISTING DESIGN Ist Stage	EXISTING DESIGN 2nd State	OPTIMUM DESIGN Ist Stage	OPTIMUM DESIGN 2nd Stage
1.	Number of teeth on pinion	46	97	34	76
	Number of teeth on gear	255	354	207	260
2.	Module	11.82*	10.74*	13.00	11.00
3.	PCD of pinion in m.m.	543.80	1042.0	442.0	836.0
	PCD of gear in m.m.	3020.0	3816.0	2691.0	2860.0
4.	Helix angle in degrees	28.75°	28.75°	23.08°	23.08°
5.	Transmission ratio	5.55	3.64	5.94	3.37

* Obtained by converting the circular pitch into equivalent module.

N.B. - The optimum design refers to initial design vector-3

TABLE 4.5

OPTIMUM DESIGN FOR EXAMPLE-3

S.No.	DESCRIPTION	EXISTING DESIGN	OPTIMUM DESIGN Ist Stage	OPTIMUM DESIGN 2nd Stage
1.	Number of teeth on pinion	Results not	43	87
	Number of teeth on gear	available	255	302
2.	Module		11.00	11.00
3.	PCD of pinion in m.m.		473.0	957.0
	PCD of gear in m.m.		2775.0	3322.0
4.	Helix angle in degrees		22.93°	22.93°
5.	Transmission ratio		5.82	3.44

N.B. - The optimum design refers to initial design vector-1

TABLE 4.6

OPTIMAL SOLUTIONS WITH DIFFERENT INITIAL DESIGN VECTORS FOR EXAMPLE-1

Starting Value of R	FIRST OPTIMAL SOLUTION		SECOND OPTIMAL SOLUTION		THIRD OPTIMAL SOLUTION	
	Area in Sq.Meters	Time in Min-Secs.	Area in Sq.Meters	Time in Min-Secs.	Area in Sq.Meters	Time in Min-secs.
FO/FP	1.48	11-34	1.50	7-16	1.47	6-01
10.0	1.44	7-51	1.40	6-46	1.44	7-50
50.0	1.46	6-53	1.43	6-55	1.48	6-10
100.0	1.51	6-43	1.44	6-19	1.46	6-47

Value of optimum solution = 1.40 sq.meter.

TABLE 4.7

OPTIMAL SOLUTIONS WITH DIFFERENT INITIAL DESIGN VECTORS FOR EXAMPLE-2

Starting Value of R	FIRST OPTIMAL SOLUTION		SECOND OPTIMAL SOLUTION		THIRD OPTIMAL SOLUTION	
	Area in Sq.Meters	Time in Min-secs.	Area in Sq.Meters	Time in Min-secs.	Area in Sq.Meters	Time in Min-secs.
FO/FP	1.69	7-10	1.65	6-14	1.58	5-36
10.0	1.66	6-41	1.58	6-12	1.50	6-21
50.0	1.66	6-44	1.59	7-08	1.72	7-04
100.0	1.55	7-11	1.59	7-13	1.54	6-09

Value of optimum solution = 1.58 sq.meter

TABLE 4.8

OPTIMAL SOLUTIONS WITH DIFFERENT INITIAL DESIGN VECTORS FOR EXAMPLE-3

Starting Value of R	FIRST OPTIMAL SOLUTION		SECOND OPTIMAL SOLUTIONS		THIRD OPTIMAL SOLUTION	
	Area in Sq.Meters	Time in Min-secs.	Area in Sq.Meters	Time in Min-secs.	Area in Sq.Meters	Time in Min-secs.
FO/FP	1.87	4-55	2.03	5-47	1.98	7-28
10.0	1.96	6-44	2.11	6-41	2.08	6-46
50.0	2.03	7-07	2.09	7-01	2.03	7-68
100.0	1.95	7-23	1.97	6-25	2.10	7-21

Value of optimum solution = 1.87 sq.meter.

COMPARISON OF RESULTS

EXAMPLE 1 :

Value of the objective function of the existing design	= 7.6 sq.m.
Value of the objective function of the optimum design	= 5.82 sq.m
Percentage Reduction in the Deck area	$= \frac{7.6 - 5.82}{7.6} \cdot 100$
	= 23.4%

EXAMPLE 2 :

Value of the objective function of the existing design	= 10.30 sq.m.
Value of the objective function of the optimum design	= 5.70 sq.m
Percentage Reduction in the Deck area	$= \frac{10.30-5.70}{10.30} \cdot 100$
	= 54.3%

EXAMPLE 3 :

In the absence of the data for this example, the results could not be compared.

CHAPTER V

CONCLUSIONS AND RECOMMENDATIONS

The application of the proposed algorithm to solve a marine gearing problem shows that the complicated machine design problem can be tackled by computer. Whether the algorithm is of any practical use or not depends upon the problem and the situation in which it is encountered. In general, the question as to whether a mathematical programming approach is suitable or not depends upon the following considerations :

1. The need for obtaining an optimum design.
2. The effort and cost required to write a programme identifying the variables and constraints in the problem compared to manhour requirements in a conventional design office for the same problem.
3. The capability of the programme to tackle a wide range of similar problems.

The programme, that has been developed, is very general and can handle any type of marine gearing design problem to suit the various requirements of the designer. Therefore it spreads out the cost of developing the programme and the computer time. As a result, the economy due to the computer aided design procedure in terms of time and engineer-hours in comparison with the conventional design procedure may be more.

The author does not claim to give sufficient information for final manufacturing of the gears, but then, the results of the programme definitely allow the designer to draw the final drawing and obtain other important dimensions from the assembly drawing itself. Thus the routine part of the analysis is not needed and using computer the designer obtains an optimum design as a bargain. Finally, it can be stated that the time and cost of developing the general programme, though considerable is quite insignificant to those for preliminary design of even a single stage reduction gearing to tailor the various geometrical and strength requirements. The added advantage of programming is that a number of optimal designs with different degrees of reliability can be obtained with least efforts. This makes room for the designer to use his discretion to pick up a suitable design.

5.1 Recommendations

The author recommends the following as a basis for further work in the field.

1. One very severe short coming of the algorithm is that it is not self starting. One has to feed starting feasible solution. However, remedying this is not as difficult as it sounds. The initial number of teeth and module can be given very high values thereby leaving no room for inadequate power transmission. But the

difficulty is likely to come in satisfying the various geometrical constraints. With few hit and trial efforts, one can find a feasible solution.

2. The programme could be made more general by incorporating the design of gear box for multiscrew ships. However, this requires an entirely different layout of the reduction gearing, but the same could be achieved from the present programme without much efforts compared to the amount of versatility the programme is likely to acquire.

3. One of the difficulties encountered was the erratic behaviour of linear minimisation algorithm at points near the constraint boundaries. It is suggested that a better way of improving the method may be to modify the linear minimisation routine, and tailor it specifically for the particular example being considered rather than making it more general for all types of problems.

4. In gradient calculations, the derivatives of constraints which do not vary with respect to certain design parameters have also to be computed for convenience in programming, though it is obvious that the derivatives would be zero. This accounts for much computer time. An algorithm may be developed in a minimisation programme which would identify those constraints which do not vary with respect to a variable about which the derivative is being calculated and skips the derivative computation.

REFERENCES

1. HADLEY, G., 'Non Linear Programming', Addison Wesley Publication.

2. FOX, R.L., 'Mathematical Methods in Optimisation', one of the series of special lectures on 'An Introduction to Structural Optimisation', University of Waterloo, 1968.

3. MOORE, I., 'Linkage Optimisation using Inequality Constraints', M.S. Thesis, Division of Solid Mechanics Structures and Mechanical Design', June 1968.

4. CINADER, F.A., 'A Mathematical Programming Approach to Design of a Transmission', M.Sc.Thesis, Division of Solid Mechanics, Structures and Mechanical Design, Case Western Reserve University, July 1970.

5. KAPOOR, M.P., 'Automated Optimum Design of Structure Under Dynamic Response Restrictions', Ph.D. Thesis, Division of Solid Mechanics, Structures and Mechanical Design, Case Western Reserve University, Dec. 1968.

6. FIACCO and McCORMIC, 'Computational Algorithm for the S.U.M.T. for Non-linear Programming' Management Science, Vol. 10n4, 1964, pp. 601-617.

7. FLETCHER, R. and POWELL, M.J.D., 'A Rapidly Convergent Descent Method for Minimization', The Computer Journal, Vol. 6, No. 2, July 1963, pp. 163-168.

8. POWELL, M.J.D., 'An Efficient Method for Finding the Minimum of a Function of Several Variables Without Calculating Derivatives', The Computer Journal, Vol. 7, No. 2, July 1964, pp. 155-162.

9. E.J. WELLAUER, 'Surface Durability of Helical and Herringbone Gears', Machine Design V36n11 May 7, 1964, pp. 156-64.

10. E.J. WELLAUER, 'Analysis of Factors used for Strength Rating of Helical Gears', ASME-TRANS-Journal of Engineering for Industry, Vol. 82, Ser.Bn3 August 1960, pp. 205-212.

11. BEALE, G.B., 'Marine Engineering and Transmission', Marine Engineering & Naval Architect, Vol.92 n 1124, Nov. 1969, pp. 467-72.

12. WELLAUER, E.J., 'Applying New Gear Strength and Durability Formula to Design', ASME - Paper 64, May, 1964.

13. WELLAUER, E.J., 'How to Predict the Wear Strength of Helical and Herringbone Gears', Machine Design, Vol. 33 n 26, Dec. 21, 1961, pp. 125-40.

14. WHITE, G., and SANGER, D.J., 'On Synthesis of Gear Trains of Minimum Size', Machine Tool Design and Research, Vol. 8, n1 March 1968, pp 27-31.

15. SAVERIN, M.M., 'Increasing the Loading on Gearing and Decreasing its Weight', Translated in English from Russian by Pergamon Institute for Department of Scientific and Industrial Research and later published in Book Form.

16. JOHNSON, R.C., 'Optimum Design of Mechanical Element', John Wiley and Sons Inc., 1961.

17. COULING, S.A., 'Industrial and Marine Gearing', John Wiley and Sons, 1962.

18. HUEBNER, G.J., 'Computer Based Selection of Balanced Life Gears', SAE Trans, Vol. 68, 1960, pp. 305-17.

19. WILLIS, R.J., 'New Equations and Charts Pickoff Lightest Weight Gears', Product Engineering Vol. 34, n2 Jan. 21, 1963, pp. 64-75.

20. TUPLIN, W.A., 'Notes on Design of Crossed Helical Gear', Machinery Vol. 105n2699, Aug. 5,1964, pp. 365-69.

21. DUDLEY, DARLEY, W., 'Practical Gear Design', McGraw-Hill Book Company, 1954.

APPENDIX A

CORRECTION OF GEAR TOOTH

The number of gear teeth which have been obtained from the optimisation programme shall have non-integer values in general. They must be rounded off to the next integer with a view that the final speeds at spindle do not change by more than 2%. From strength considerations, if the pinions are given positive corrections, their strength increases at the cost of already overdesigned gears. Total correction X_T required to accommodate the modified number of teeth is given as

$$X_T = \frac{(INV(\alpha)-INV(\alpha_0)) \cdot (Za' + Zb')}{2 \tan(\alpha_0)}$$

where Za' = modified number of teeth in pinion

Z_b' = modified number of teeth in gear

α_0 = modified pressure angle

$= \tan^{-1}\left(\frac{1-Q^2}{Q}\right)$

and $Q = \frac{Z_a' + Z_b'}{Z_a + Z_b} \cdot \tan\alpha_0$

where Z_a = unmodified number of teeth in pinion

Z_b = unmofied number of teeth in gear.

APPENDIX B

COMPUTER PROGRAMME

This Appendix gives the descriptions of nine subroutines which add up to make the optimised design of marine gearing. A listing of each subroutine is included. The input and output formats are described and a part of output listing is printed.

B.1 The Programme

The programming has been split in two phases, optimisation and design. The first part is carried out in the main routine which requires another six subroutines to reach its end. Second phase is carried out in the subroutines Modif, Design and Correc, which needs the optimum design vector from main as input information.

The purpose of different subroutines together with the necessary input for them follows in the subsequent paragraphs.

B.1-1 Main :

Purpose - To read the input data to extrapolate the design vector for a new starting point (to be used in subroutine Uncons.) and to stop iterations when convergence criteria are satisfied.

B.1-2 Subroutine Const. :

Purpose - To calculate the various expressions used to calculate the constraints in the subroutine 'Penal', and to design the gear shafts.

B.1-3 Subroutine Penal(FO,FP,FT,X,N,R,GX) :

Purpose - To calculate the value of the objective function and to evaluate the normalised constraints. To set the value of penalty function as 10^{30} whenever a constraint is violated.(This identifies an acceptable solution from the unacceptable one).

Input - The present design vector, number of variables and the value of r in the unconstrained minimisation are the inputs to this subroutine. For every evaluation of constraint the subroutine first calls the subroutine 'Const' to modify different expressions, used in the constraint evaluation.

B.1-4 Subroutine UnCons.(X,N,R.):

Purpose - To carryout unconstrained minimisation for any 'r' value in the sequence of r minimisations. In process, to identify a situation as whether to modify H matrix or not and to stop iterations when the convergence criterion are satisfied.

Input - This subroutine is called from Main every time when 'r' is reduced. Starting design vector, number of design variables, and the value of r are the input information. This subroutine uses three other subroutines namely PENAL, SEARCH, HMATRI at different stages. Convergence limits are transferred from Main through common statements.

B.1-5 Subroutine Search (STEP, X,R,N) :

Purpose - To carryout the one dimensional minimisation and evaluate a step size which makes the penalty function minimum in the present direction of move. This subroutine is called from Main and 'Unconst' each time a new direction is found out.

Input - The new direction and the present value of design vector are input here. Convergence criterion has been permanently defined in the subroutine itself.

B.1-6 Subroutine Grad (G,X,N,R) :

Purpose - To calculate the gradient at any point in the design space by forward interpolation. This is called from Uncons every time a step in the old direction is taken and a new design vector is reached.

Input - Inputs are the present value of $\overline{X}$ and number of design variables. The step length used in the forward difference scheme is permanently defined in the subroutine.

B.1-7 Subroutine HMATRI (H, SIG, N) :

Purpose - To modify the Hessian matrix used to calculate the direction of move at every point. Every time step size in the previous direction meets the convergence requirements, this subroutine is called from Uncons.

Input - Inputs for the subroutine are the present Hessian matrix, the value of the step in the previous move, the difference of the present and previous gradient vector and the number of variables.

B.1-8 Subroutine MODIF (X,N,R) :

Purpose - To round off the number of teeth of different gears and pinions to make them integers. To round off the module of gears to the next standard module available.

Input - The final optimum design vector.

B.1-9 Subroutine CORREC (ZP, ZG, TRNS, XT, XP, XG):

Purpose - To distribute the total correction on the pinion and gear for achieving the compatibility. The information about the graphs in I.S.(3756) used for this purpose is permanently

defined in the subroutine in data statement. This subroutine is called for correction of each pair of gears.

Input - The total correction value obtained from the subroutine DESIGN, rounded number of teeth in gears and pinions and the transmission ratio between them.

B.1-10 Subroutine DESIGN (XO, N, NC) :

Purpose - To round off the centre distance between the gears. To calculate the total correction for different sets of gears and to convert FPS system into MKS system.

Input - The final design vector from subroutine MODIF.

B.2 Programme Output :

The input data is written out in the same format in which it is read.

Other outputs are presented with self explaining headings.

B.3 Programme Listing and Sample Output :

The subsequent pages contain the listing of all the subroutines and a portion of output.

```
$IBJOB
$IBFTC  MAIN
        DIMENSION  XMIN(15), XN(15), SPAN(15), X1(15)
        DIMENSION GX(40), GX1(40), GX2(40), D(15), O(15)
        DIMENSION  X(15), XR(15), XL(15), XO(15), G(15)
        COMMON/ DMS1/ PI, RAD, NC
        COMMON/ DMS2/ CO1, CR1, CL1, CS1, CH1, CT1, CF1
         COMMON/DMS3/CO2,CR2,CL2,CS2,CH2,CT2,CF2
        COMMON/ DMS4/ HP, PHIAN, EG, UG, RPM1, RPM2, RPM3
        COMMON/DMS5/DP1,DP2,DG1,DG2,B1,B2,ZG1,ZG2,PLV1,PLV2,COSB
        COMMON/ DMS 6/ C1, C3, AK, GO, CF, SENS, N
        COMMON/ DMS7 / SALB1, SALB2, FATS, SHEAR
        COMMON/ DMS 8/ CD1, CD2, CD3,OZP1,OZG1, OZP2, OZG2
        COMMON/ DMS9/ TOR1, TOR2, TOR3, BM1, BM2, BM3
        COMMON/DMS10/ EPS1, EPS2, EPS3, EPS4, ITIR
        COMMON/ DMS11/ ZP1, ZP2, PND1, PND2, SIAD1, SIAD2
        COMMON/ DMS12/ SIA1, SIA2, TRAN1, TRAN2, ALPHD, ALPHA
        COMMON/ DMS 13/ CNF1, CNF2, ERR, DSH1, DSH2, DSH3, SC1, SC2
        COMMON/ DMS 14/ XMIN, XN, SOAN
        COMMON/ DMS 15/ GX, GX1, GX2
        COMMON/ DMS16/ HPW1, HPW2, +PB1, HPB2, HEIGHT, BASE, AREA
        COMMON/DMS18/ CORRP1, CORRG^, CORRP2, CORRG2
        COMMON/ DMS 17/ D, Y, G
        COMMON/ DMS 20 / FCR1, FCR2, PCR1, PCR2, WW1, WW2, WD1, WD2
        COMMON/DMS 19/ PMOD1, PMOD2, TRAN,       FOB
 16     FORMAT( /, 10X, * EXTRAPOLATED VALUES OF VARIABLES AT NEXT
      1 R- VALUES*,/, 3( 10X, 5F10.6 / ))
 23     FORMAT( ///, 10X, * NORMALISED OPTIMUM DESIGN VARIABLES*,/,
      1 3( 10X, 5F10.6 / ))
 25     FORMAT( /, 10X, * ACTUAL DESIGN VARIABLES*, /, 3(10X,5F10.4 / )))
 27     FORMAT( / 10X,* DIAMETER OF FIRST PINION=*, F12.6,//10X,* DIAMETE
      1R OF FIRST GEAR=*,F12.6 )
 28     FORMAT( / 10X,* DIAMETER OF SECD. PINION=*, F12.6,//10X,* DIAMETE
      1R OF SECD. GEAR=*,612.6 )
 29     FORMAT(  / 10X,* THE FACE WIDTH OF FIRST GEAR SET=*,F12.6,
      1  //10X,* THE FACE WIDTH OF SECD.GEAR SET=*,F12.6)
 31     FORMAT( / 10X,* THE PITCH CONTACT RATIO OF FIRST GEAR SET=*,F12.6
      1,// 10X,* THE FACE CONTACT RATIO OF FIRST GEAR SET=*,F12.6 )
 32     FORMAT( / 10X,* THE PITCH CONTACT RATIO OF SECD. GEAR SET=*,F12.6
      1,// 10X,* THE FACE CONTACT RATIO OF SECD. GEAR SET=*,F12.6 )
 33     FORMAT( / 10X,* THE PITCH LINE VELOCITY OF FIRST GEAR SET=*,F15.6
      1,                    // 10X,* THE PITCH LINE VELOCITY OF SECD. GEAR
      2 SET= *,F15.4)
 34     FORMAT( / 10X,* T8E MAXIMUM H.P OF THE FIRST  GEAR SET ON WEAR BA
      1SIS=*,F15.4,// 10X,* THE MAXIMUM H.P. OF THE FIRST GEAR SET ON BEN
      2DING STRENGTH BASIS=*, F15.4 )
 35     FORMAT( / 10X,* THE MAXIMUM H.P OF THE SECD.  GEAR SET ON WEAR BA
      1SIS=*,F15.4,// 10X,* THE MAXIMUM H.P. OF THE SECD. GEAR SET ON BEN
      2DING STRENGTH BASIS=*, F15.4 )
 36     FORMAT( / 10X,* NUMBER OF TEETH OF FIRST PINION=*,F12.6,//10X,*N
      1UMBER OF TEETH OF SECD. PINION=*, F12.6 )
 37     FORMAT( / 10X,* NUMBER OF TEETH OF FIRST GEAR  =*,F12.6,//10X,*5X
```

```
     1UMBER OF TEETH OF SECD. GEAR=*,F12.6 )
38     FORMAT( / 10X,* T8E DIAMETERAL PITCH OF FIRST GEAR SET=*,F12.5,//
     110X,* THE DIAMETERAL PITCH OF SECD. GEAR SET=*, F12.5 )
39     FORMAT( / 10X,* HELIX ANGLE FOR THE FIRST GEAR SET=*,F12.5,* DEG*
     1EES*, // 10XT* HELIX ANGLE FOR THE SECD. GEAR SET=*,F12.5,* DEGREE
     2S*)
40     FORMAT( / 10X,* THE TRANSMISSION RATIO IN THE FIRST REDUCTION=*,,
     1F12.5,// 10X,* THE TRANSMISSION RATIO IN THE SECD. REDUCTION=*,F55
     2.5 )
41     FORMAT( / 10X,* T8E PERMISSIBLE ERROR IN THE MANUFACTURE=*=F12.')
44     FORMAT(/10XT * INITIAL VALUE OF R= *, F10.2)
65     FORMAT( // 10X, * THESE ATE THE VALUES FOR CO1, CR1, CL1,  CS1,
     1 CH1, CT1, CF1 FACTORS * )
66     FORMAT(  / 10X, * THESE ARE THE VALUES FOR CO2, CR2, CL2, CS2,
     1 CH2, CT2, CF2 FACTORS * )
67     FORMAT(// 10X, *THESE ARE THE MODIFIED OPTIMUM DESIGN VARIABLES**
68     FORMAT( 10X, 47(1H-))
70     FORMAT(  8F10.5 )
71     FORMAT ( 7F6.2 )
72     FORMAT ( I3 )
73     FORMAT(  / 10X,* THESE ARE THE VALUES FOR CF, SENS. FACTORS*  )
74     FORMAT (  / 10X, 11F 10.5 )
75     FORMAT(  / 10X, * THESE ARE THE VALUES FOR THE DESIGN VARIABLES*)
76     FORMAT(  / 10X, * THESE ARE THE VALUES FOR THE MINIMUM VALUES
     1 FOR THE DESIGN VARIABLES *)
77     FORMAT(  / 10X, * THESE ARE THE VALUES FOR THE SPAN OF THE DESIGN
     1 VARIABLES *)
78     FORMAT(  / 10X, * NO  OF  ITIRATIONS= * I3 )
80     FORMAT(/  10X,* THESE ARE THE VALUES FOR EPS1, EPS2, EPS3, EPS4*)
91     FORMAT(/// 10X,*INPUT DATA *)
92     FORMAT(10X, 10(1H-))
138    FORMAT(/ 10X,* THE MODULE FOR FIRST GEAR SET =*,F12.4,// 1 X,* TH
     1E MODULE FOR THE SECD. GEAR SET =*,F12.4 )
209    FORMAT( /// 10X, *OPTIMUM GEAR DESIGN *)
210    FORMAT(  10X, 20(1H-))
211    FORMAT(  / 29X,* FIRST PINION *,2X,* FIRST GEAR *,4X,* SECD. PINI
     1ON *,4X,* SECD. GEAR */  )
212    FORMAT( /  10X,* NUMBER OF TEETH      *,I6   ,  9X,I6   ,  6X,IX,
     1 5X,I6 )
213    FORMAT(  / 10X, *P C D IN M M.        *,F10.2, 5X, F10.2,  5X,F10
     1.2,   5X,F10.2  )
214    FORMAT(  / 10X, *FACE WIDTH IN M M.   *,F10.2,  5X,F10.2,  5X,
     1 F10.2,  5X, F10.2  )
215    FORMAT(  / 10X, *MODULES IN M.M.      *,F10.2,  5X,F10.2,  5X,
     1F10.2,  5X, F10.2  )
216    FORMAT(  / 10X, *HELIX ANGLE IN DECREE*,F10.2,  5X,F10.2,  5X,
     1 F10.2,  5X, F10.2  )
217    FORMAT( /  10X, *PROFILE CONTACT RATIO*,F10.2,  5X, F10.2,  5X,
     1 F10.2,  5X, F10.2  )
218    FORMAT(  / 10X, *FACE CONTACT RATIO   *, F10.2,  5X, F10.2,  5X,
     1 F10.2,  5X, F10.2  )
219    FORMAT(  / 10X, *P L VELOCITY IN M P S*,F10.2,  5X, F10.2,  5X,
```

```
    1 F10.2,   5X, F10.2   )
220   FORMAT(  / 10X, *HORSE POWER IN WEAR   *, F12.2,   5X, F12.2=  5X,,
    1 F12.2,   5X, F12.2   )
221   FORMAT( /  10X, *HORSE POWER IN BENDING*,F12.2,  5X, F12.2=  5X,
    1 F12.2,   5X, F12.2   )
222   FORMAT( /  10X, *TRANSMISSION- RATIO   *, F10.2,  5X, F10.2,  5X,
    1 F10.2,   5X, F10.2   )
223   FORMAT( /  10X, *ALLOW MANF ERR IN M M.*,F10.4,  5X, F10.4,  5X,3
    1 F10.4,   5X, F10.4   )
224   FORMAT( / 10X,  *CORREC FACTOR         *,F10.2,  5X,F10.2,  5X,
    1F10.2,  5X, F10.2   )
231   FORMAT(/// 10X, *GEOMETRY OF GEAR SET * )
232   FORMAT( 10XT 20(1H-))
233   FORMAT(/  10X, * BASE= *,F10.2, * M M.   *)
234   FORMAT( /  10X, * HEIGHT =*, F10.2, * M M.    *)
235   FORMAT( / 10X,* AREA= *, F10.2,* SQ. METS *)
241   FORMAT( /// 10X,* INSPECTION DATA * )
242   FORMAT(  10X, 16( 1H- ))
243   FORMAT(/// 10X, *FIRST PINION *,  5X,*FIRST GEAR *,  5X,* SECD.
    1 PINION *,  5X, * SECD. GEAR *  )
300   FORMAT(//10X,*INPUT DESIGN DATA*)
350   FORMAT(10X,18(1H-))
301   FORMAT(/ 10X,*HORSE POWER=*,F10.3 )
302   FORMAT(/ 10X,*TURBINE SPEED=*,F8.1,*RPM*)
303   FORMAT(/ 10X,*PROPELLER SPEED=*,F8.3,*RPM*)
304   FORMAT(/ 10X,*OVERALL REDUCTION RATIO=20.0*)
500   FORMAT(//10X,* OBJECT FUNCTION *,F10.3,//,10X,* PENALTY FUNCTION
    1 *, F10.3,//,10X,* TOTAL FUNCTION *, F10.3 )
501   FORMAT( 10X, * DESIGN VARIABLES *, / , 3(10X, 6F16.6 / ))
502   FORMAT( 10X, * CONSTRAINTS *,//, 10( 10X, 8F10.5 /))
503   FORMAT( 10X, * GRADIENT VECTOR*,//, 3( 10X, 5F10.6 / ))
      READ 71, CO1, CR1, CL1, CS1, CH1, CT1, CF1
      READ 71, CO2, CR2, CL2, CS2, CH2, CT2, CF2
      N= 11
      NC= 36
      PI= 4.0*ATAN( 1.0 )
      RAD= PI/ 180.0
      HP= 7.5*10.0**4
      PHIAN= 20.0* RAD
      RPM1= 3000.0
      EG= 29.0*10.0**6
      C3= 1.67
      AK= 861.0
      GO= 1.18*10C0**6
      SALB1=  2.25*10.0**5
      SALB2=  2.25*10.0**5
      UG= 0.20
      FATS= 1.5*10.0**5
      SHEAR= 8.5*10.0**4
```

```
      ZP1= 100.0
      ZP2= 75.0
      SIAD1= 30.0
      SIAD2= 30.0
      PND1= 3.85
      PND2= 2.16
      TRAN1= 5.0
      ALPHD= 65.0
      CNF1= 11.5
      CNF2= 8.0
      ERR= 0.00085
      ALPHA= ALPHD*RAD
      SIA1= SIAD1* RAD
      SIA2= SIAD2* RAD
      C1= C3*10.0**6*ERR
      TRAN2= 20.0/ TRAN1
C     ******************************,*********************************
C
C     THE DESIGN VARIABLES ARE NORMALISED AS UNDER
C
C     ****************************************************************
      READ 70, CF, SENS
      READ 70, ( XMIN(I), I= 1, N )
      READ 70, (X1(I), I= 1, N )
      READ 70, ( SPAN(I), I= 1, N)
      READ 70, EPS1, EPS2, EPS3, EPS4
      READ 72, ITIR
      PRINT 91
      PRINT 92
      PRINT 65
      PRINT 74, CO1, CR1, CL1, CS1, CH1, CT1, CF1
      PRINT 66
      PRINT 74, CO2, 3R2, CL2, CS2, CH2, CT2, CF2
      PRINT 73
      PRINT 74, CF, SENS
      PRINT 76
      PRINT 74, ( XMIN(I), I= 1, N )
      PRINT 75
      PRINT 74, ( X1(I), I= 1, N)
      PRINT 77
      PRINT 74, ( SPAN(I), I= 1, N )
      PRINT 80
      PRINT 74, EPS1, EPS2, EPS3, EPS4
      PRINT 78, ITIR
      DO 505 I= 1, N
```

```
50      X1(I)= (X1(I)- XMIN(I))/ SPAN(I)
        PRINT 79
        PRINT 74, ( X1(I), I= 1, N )
        CALL PENAL( FO, FP, FT, X1, N, 1.0, GX )
        DO 43 I= 1, NC
43      IF( GX(I). LT.0.2) FP= FP+5.0-1.0/ GX(I)
        PRINT 500, FO, FP, FT
        PRINT 502, ( GX(I), I= 1, NC)
        R= 0.0617
        R= 5.0
        PRINT 44, R
        L3= 0
1       CALL UNCONS ( X1, N, R )
        DO 5 I= 1, N
5       XN(I)= X1(I)
        CALL PENAL( FO, FP, FT, XN, N, R, GX )
        PRINT 500, FO, FP, FT
        PRINT 501, ( XN(I), I= 1, N )
        PRINT 502, ( GX(I), I= 1, NC )
        CALL GRAD( XN, N, G, R )
        PRINT 503, ( G(I), I= 1, N )
        L3= L3+1
        IF( L3. EQ. 1 ) GO TO 6
        IF( ABS((FO- FLO)/FO ) . LT. EPS4) GO TO 50
9       SS= ABS(( FT-FL)/FL)
        IF( SS. GT. EPS1) GO TO 3
        DO 10 I= 1, N.
        A1= ABS(( XN(I)- XL(I))/ XL(I))
        IF( A1. GT. EPS2) GO TO 3
10      CONTINUE
        DO 11 I= 1, N
11      XO(I)= ( CF**0.5*XN(I)-XL(I))/(CF**0.5- 1.0 )
        CALL PENAL( FO, FP, FT1, XO, N, R, GX )
        IF( FT1. GT. 10.0E+20) GO TO 3
        IF( FT1. GT. FT) GO TO 3
        SS= ABS(( FT1- FT)/ FT)
        IF( SS. GT. 0.001  ) GO TO 3
        DO 12 I= 1, N
        IF( A1. GT. 0.001 ) GO TO 3
        A1= ABS(( XO(I)- XN(I))/ XN(I))
12      CONTINUE
        GO TO 26
6       RL= R
        DO 7 I= 1, N
7       XL(I)= XN(I)
        FLO= FO
        FL= FT
        R= R/CF
        DO 8 I= 1, N
8       X1(I)= XN(I)
        IF( R. LE. 10.0E-3 ) GO TO 50
        GO TO 1
3       RL= R
```

```
      FL= FT
      FLO= FO
      R= R/CF
      DO 15 I= 1, N
      XL(I)= XN(I)
15    XR(I)= XN(I)+1.0/ CF**0.5*(XN(I)-XL(I))
      PRINT 16, ( XR(I), I= 1, N )
      CALL PENAL( FO, FP, FT2, XR, N, R, GX )
      IF( FT2. GT. 10.0E+2 ) GO TO 18
      IF( FT2. GT. FL ) GO TO 18
      DO 19 I= 1, N
19    X1(I)= XR(I)
      IF( R. LE. 10.0E-5 ) GO TO 50
      GO TO 100
18    DO 21 I= 1, N
      XL(I)= XN(I)
21    X1(I)= XN(I)
      GO TO 1
50    CONTINUE
      DO 51 I= 1, N
51    XO(I)= X1(I)
26    PRINT 23, ( XO(I), I= 1, N )
      DO 24 I= 1, N
24    XO(I)= XO(I)*SPAN(I)+ XMIN(I)
      PRINT 25, ( XO(I), I= 1, N )
      DO 30 I= 1, N
30    XO(I)= ( XO(I)- XMIN(I))/ SPAN(I)
      CALL PENAL( FO, FP, FT, XO, N, R, GX )
      PRINT 27, DP1, DG1
      PRINT 28, DP2, DG2
      PRINT 29, B1, B2
      PRINT 31, PCR1, F3R1
      PRINT 32, PCR2, FCR2
      PRINT 33, PLV1, PLV2
      PRINT 34, HPW1, HPB1
      PRINT 35, HPW2, HPB2
      PRINT 36, ZP1, ZP2
      PRINT 37, ZG1, ZG2
      PRINT 38, PND1, PND2
      PRINT 39, SIAD1, SIAD2
      PRINT 40, TRAN1, TRAN2
      PRINT 41, ERR
      CALL MODIF( XO, N, NC, R )
      PRINT 67
      PRINT 68
      CALL DESIGN( XO, N, NC )
      PRINT 27, DP1, DG1
      PRINT 28, DP2, DG2
      PRINT 29, B1, B2
      PRINT 31, PCR1, FCR1
      PRINT 32, PCR2, FCR2
      PRINT 33, PLV1, PLV2
```

```
      PRINT 34, HPW1, HPB1
      PRINT 35, HPW2, HPB2
      PRINT 36, ZP1, ZP2
      PRINT 37, ZG1, ZG2
      PRINT 138, PMOD1, PMOD2
      PRINT 39, SIAD1, SIAD2
      PRINT 40, TRAN1, TRAN2
      PRINT 41, ERR
      PRINT 300
      PRINT 350
      PRINT 301, HP
      PRINT 302, RPM1
      PRINT 303, RPM3
      PRINT 304
      PRINT 209
      PRINT 210
      PRINT 211
      IZP1= ZP1
      IZP2= ZP2
      IZG1= ZG1
      IZG2= ZG2
      PRINT 212, IZP1, IZG1, IZP2, IZG2
      PRINT 213, DP1, DG1, DP2, DG2
      PRINT 214, B1, B1, B2, B2
      PRINT 215, PMOD1, PMOD1, PMOD2, PMCD2
      PRINT 224, CORRP1, CORRG1, CORRP2, CORRG2
      PRINT 216, SIAD1, SIAD1, SIAD2, SIAD2
      PRINT 217, PCR1, PCR1, PCR2, PCR2
      PRINT 218, FCR1, FCR1, FCR2, FCR2
      PRINT 219, PLV1, PLV1, PLV2, PLV2
      PRINT 220, HPW1, HPW1, HPW2, HPW2
      PRINT 221, HPB1, 8PB1, HPB2, HPB2
      PRINT 222, TRAN1, TRAN1, TRAN2, TRAN2
      PRINT 223, ERR, ERR, ERR, ERR
      PRINT 231
      PRINT 232
      PRINT 233, BASE
      PRINT 234, HEIGHT
      PRINT 235, AREA
      STOP
      END
```

```
$IBFTC CONST
      SUBROUTINE CONST
      DIMENSION  XMIN(15), XN(15), SPAN(15), X1(15)
      COMMON/ DMS1/ PI, RAD, NC
      COMMON/ DMS2/ CO1, CR1, CL1, CS1, CH1, CT1, CF1
      COMMON/ DMS3/ CO2, CR2, CL2, CS2, CH2, CT2, CF2
      COMMON/ DMS4/ HP, PHIAN, EG, UG, RPM1, RPM2, RPM3
      COMMON/DMS5/DP1,DP2,DG1,DG2,B1,B2,ZG1,ZG2,PLV1,PLV2,COSB
      COMMON/ DMS 6/ C1, C3, AK, GO, CF, SENS, N
```

```
      COMMON/ DMS7 / SALB1, SALB2, FATS, SHEAR
      COMMON/ DMS 8/ CD1, CD2, CD3,OZP1,OZG1, OZP2, OZG2
      COMMON/ DMS9/ TOR1, TOR2, TOR3, BM1, BM2, B-3
      COMMON/DMS10/ EPS1, EPS2, EPS3, EPS4, ITIR
      COMMON/ DMS11/ ZP1, ZP2, PND1, PND2, SIAD1, SIAD2
      COMMON/ DMS12/ SIA1, SIA2, TRAN1, TRAN2, ALPHD, ALPHA
      COMMON/ DMS 13/ CNF1, CNF2, ERR, DSH1, DSH2, DSH3, SC1, SC2
      COMMON/ DMS 14/ XMIN, XN, SPAN
      COMMON/ DMS16/ HPW1, HPW2, HPB1, HPB2, HEIGHT, BASE, AREA
      COMMON/DMS18/ CORRP1, CORRG1, CORRP2, CORRG2
      COMMON/DMS 19/ PMOD1, PMOD2, TRAN,       FOB
      COMMON/ DMS 20 / FCR1, FCR2, PCR1, PCR2, WW1, WW2, WD1, WD2
      TRAN2= 20.0/ TRAN1
      DP1= ZP1/( PND1*COS(SIA1))
      DP2= ZP2/(PND2* COS(SIA2))
      B1= CNF1*PI*COS(SIA1)/ PND1
      B2= CNF2* PI* COS(SIA2)/ PND2
      DG1= DP1* TRAN1
      DG2= DP2* TRAN2
      RPM2= RPM1/ TRAN1
      RPM3= 150.0
      TOR1=  63000.0*HP/ RPM1
      TOR2=  63000.0*HP/ RPM2
      TOR3=  63000.0*HP/ RPM3
      BM1= 0.50* TOR1
      BM2= 0.50* TOR2
      BM3= 0.50* TOR3
C*****************,*****************************************************
C
C     CALCULATION OF PITCH CONTACT RATIO AND FACE CONTACT RATIOS
C
C**********************************************************************
      PCR1= 0.555*TRAN1**0.032* ZP1**0.104* COS(SIA1)**1.504
      FCR1= CNF1* SIN(SIA1)* COS(SIA1)
      PCR2= 0.555* TRAN2**0.032* ZP2**0=104* COS(SIA2)**1.504
      FCR2= CNF2*SIN(SIA2)* COS(SIA2)
      PLV1= PI*DP1*RPM1/ 12.0
      PLV2= PI*DP2* RPM2/ 12.0
      WT1= 33000.0*HP/ PLV1
      WT2= 33000.0* HP/ PLV2
      IF(PLV1. GT. 5000.0)CV1= SQRT(78.0/(78.0+SQRT(PLV1)))
      IF( PLV1. LE. 5000.0) CV1= 78.0/(78.0+SQRT(PLV1))
      IF( PLV2.GT.5000.0) CV2= SQRT(78.0/(78.0+SQRT(PLV2)))
      IF( PLV2.LE.5000.0) CV2= 78.0/(78.0+SQRT(PLV2))
600   FORMAT( // 10X, 3F15.5 )
      BMAX1= SQRT(2.0*WT1/(ERR*GO))
      BMAX2= SQRT( 2.0*WT2/( ERR*GO))
      IF(BMAX1.GT.B1) CM1= BMAX1/(BMAX1-0.5*B1)
      IF( BMAX2.LE.B2) CM2= 2.0*B2/ BMAX2
      IF( BMAX2.GT.B2) CM2= BMAX2/(BMAX2-0.5*B2)
      GF1= 0.450*TRAN1/(TRAN1+1.0)
      GF2= 0.450*TRAN2/(TRAN2+1.0)
```

```
      CP1= SQRT(EG/(2.0*PI*(1.0-UG*UG)))
      CP2= CP1
      SC1= CP1*SQRT(WT1*CO1*CS1*CM1*CF1/(CV1*B1*DP1*GF1))
      SC2= CP2* SQRT(WT2*CO2*CS2*CM2*CF2/(CV2*B2*DP2*GF2))
      SCDP1= (SALB1*DP1*CL1*CH1/(CP1*CT1*CR1))**2
      SCDP2= (SALB2*DP2*CL2*CH2/(CP2*CT2*CR2))**2
      HPW1= RPM1*B1*GF1*CV1*SCDP10(12600(.0*CS1*CM1*CF1*CO1)
      HPW2= ( RPM2*B2/1260^0.0)* SCDP2/((S2*CM2*CF2*C(2)* CV2
      WDT1= (B1*C1*C6S(SIA1)*COS(SIA1)+WT1)
      WDT2= (B2*C1*COS(SIA2)*COS(SIA2)+ WT2)
      WD1= WT1+(0.05*PLV1*WDT1*COS(SIA1))/(SQRT(WDT1)+0.05*PLV1)
      WD2= WT2+0.05*PLV2*WDT2*COS(SIA2)/(0.05*PLV2+SQRT(WDT2))
      Q1= 2.0*TRAN1/(TRAN1+1.0)
      Q2= 2.0*TRAN2/(TRAN2+1.0)
      WW1= PCR1*DP1*B1*AK*Q1/(COS(SIA1)*COS(SIA1))
      WW2= PCR2*DP2*B2*AK*Q2/(COS(SIA2)*COS(SIA2))
      YP1= 0.550-1.41/(ZP1/(COS(SIA1)**3)- 1.89)**0.756
      YP2= ^.550-1.41/(ZP2/(COS(SIA2)**3)-1.89)**0.756
      HPB1= (RPM1**0.787*TRAN1**0.040/74000.0)*(FATS**1.26/C3**0.067)**
     1 (DP1**3.224*CNF1**1.92*YP1**1.26/(COS(SIA1)**2.004*(ZP1/COS(SIA1
     2 )**3)**2.450))
      HPB2= (RPM2**0.787*TRAN2**0.040/74000.0)*(FATS**1.26/C3**0.067)**
     1 (DP2**3.224*CNF2**1.192*YP2**1.26/(COS(SIA2)**2.004*(ZP2/COS(SIA2
     2 )**3)**2.450))
C     **********M******************,*******************************
C
      DSH1= (16.0*SQRT(BM1**2+TOR1**2)/(PI*SHEAR))**0.333
      DSH2= (16.0*SQRT(BM2**2+TOR2**2)/(PI*SHEAR))**0.333
      DSH3= (16.0*SQRT(BM3**2+TOR3**2)/(PI*SHEAR))**0.333
      RETURN
      END



$IBFTC FIBO
      SUBROUTINE SEARCH( STEP, XO, N, R '
      COMMON/ DMS 17/ D, Y, G
      DIMENSION XO(15),D(15),XM(15),X1(15),X2(15),GX(40),X(15)
      DIMENSION Y(15), G(15)
      INTEGER FN(12)
      DATA FN/1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144/
      DO 101 I = 1,N
  101 X(I) = XO(I)
      DIST= 0.10
    1 SUB = DIST/21=0
      LOGICAL ONE, TWO
      ONE = .FALSE.
      TWO = .FALSE.
      DO 20 ITER=1,5
      NTEMP = 8 - ITER
      ND = FN(NTEMP)
```

```
      ALFA1 = SUB*FLOAT(ND)
      ALFA2 = SUB*FLOAT(ND-1)
      DO 5 I=1,N
      X1(I) = XO(I) + ALFA1*D(I)
    5 X2(I) = XO(I) + ALFA2*D(I)
      IF(ONE) GO TO 6
      CALL PENAL(FP, FO, F1, X1, N, R, GX)
    6 IF(TWO) GO TO 7
      CALL PENAL(FO, FP, F2, X2, N, R, GX )
    7 CONTINUE
      PRINT 900, ITER, F1, F2
      IF (F1.LT.1.0E+20) GO TO 9
      IF (F2.LT.1.0E+20) GO TO 8
      DIST = ALFA2
      GO TO 1
 8     DIST = ALFA1
      GO TO 1
9     IF (ITER.EQ.5) GO TO 25
  900 FORMAT(I5, 2E20.8)
      IF(F1 .GT. F2) GO TO 15
      DO 10 I=1,N
   10 XO(I) = X2(I)
      F2 = F1
      TWO = .TRUE.
      ONE = .FALSE.
      GO TO 20
   15 F1 = F2
      ONE = .TRUE.
      TWO = .FALSE.
20     CONTINUE
   25 FMIN = AMIN1UF1, F2)
      IF(FMIN .EQ. F2) GO TO 35
      DO 30 I=1,N
   30 XM(I) = X1(I)
      GO TO 45
   35 DO 40 I=1,N
   40 XM(I) = X2(I)
45    STEP = (XM(1)-X(1))/D(1)
      DO 50 I=1,N
50    XO(I) = X(I)
      RETURN
      END



$IBFTC PENAL
      SUBROUTINE PENAL( FO, FP, FT, X, N, R, GX )
      DIMENSION  X(15), GX(40), XMIN(15), X1(15), SPAN(15), XN(15)
       COMMON/ DMS1/ PI, RAD, NC
       COMMON/ DMS2/ CO1, CR1, CL1, CS1, CH1, CT1, CF1
       COMMON/ DMS3/ CO2, CR2, CL2, CS2, CH2, CT2, CF2
```

```
      COMMON/ DMS4/ HP, PHIAN, EG, UG, RPM1, RPM2, RPM3
      COMMON/DMS5/DP1,DP2,DG1,D+2,B1,B2,ZG1,ZG2,PLV1,PLV2,COSB
      C6MMON/DMS7/ SALB1, SALB2, FATS, SHEAR
      C64MON/ DMS 8/ CD1, CD2, CD3,OZP1,OZG1, OZP2, OZG2
      C6MMON/ DMS9/ TOR1, TOR2, TOR3, BM1, BM2, BM3
      C6MMON/DMS10/ EPS1, EPS2, EPS3, EPS4, ITIR
      C6MMON/ DMS11/ ZP1, ZP2, PND1, PND2, SIAD1, SIAD2
      C6MMON/ DMS12/ SIA1, SIA2, TRAN1, TRAN2, ALPHD, ALPHA
      C6MMON/ DMS 13/ CNF1, CNF2, ERR, DSH1, DSH2, DSH3, SC1, SC2
      C6MMON/ DMS 14/ XMIN, XN, SPAN
      C6MMON/ DMS16/ HPW1, HPW2, HPB1, HPB2, HEIGHT, BASE, AREA
      C6MMON/DMS18/ CORRP1, CORRG1, CORRP2, CORRG2
      C6MMON/DMS 19/ PMOD1, PMOD2, TRAN,     FOB
      C6MMON/ DMS 20 / FCR1, FCR2, PCR1, PCR2, WW1, WW2, WD1, WD2
C     *,*****************************M***********************************
C
C     CALCULATION OF CENTER DISTANCES ANI OBJECTIVE FUNCTION
C
C     *,*****************************************************************
      DO 90 I= 1, N
 90   X1(I)= X(I)
      DO 91 I= 1, N
 91   X(I)= X(I)* SPAN(I)+ XMIN(+)
      ZP1= X(1)
      ZP2= X(2)
      PND1= X(3)
      PND2= X(4)
      SIAD1= X(5)
      SIAD2= X(6)
      TRAN1= X(7)
      ALPHD= X(8)
      CNF1= X(9)
      CNF2= X(10)
      ERR= X(11)
      CALL CONST
      TRAN2= 20.0/ TRAN1
      ZG1= OP1* TRAN1
      ZG2= OP2* TRAN2
      SENS= 0.50
      CD1= DP1*(TRAN1+1.0)/2.0
      CD2= DP2*(TRAN2+1.0)/2.0
      CD33= (CD1**2)+(CD2**2)- 2.0*CD1*CD2*COS(ALPHA)
      CD3= SQRT(CD33)
      COSB= ((CD1**2)+(CD3**2)-(CD2**2))/(2.0*CD1*CD3)
      BETA= ARCOS(COSB)
      BETAR= BETA/ RAD
      BASE= CD3
      HEIGHT= CD1* SIN(BETAR )
      AREA= CD3*CD1* SIN(BETA )/(2.0* 144.0 )
      FO= ZP1*ZP2*(TRAN1+1.0)*(T-AN2+1.0)*SIN(ALPHA)/(PND1*PND2*COS(SIA
     1 1)*COS(SIA2)*2.0)
```

```
C
C     THE FOLLOWING ARE THE CONSTRAINTS FOR THE PROBLEM
C
C
      GX(1)=  1.0- 13.7/ ZP1
      GX(2)= 1.0- 13.7/ ZP2
      GX(3)= 1.0-1.0/FCR1 + 0.8*PCR1/ FCR1
      GX(4)= 1.0-1.0/FCR2 + 0.8*PCR2/FCR2
      GX(5)= 1.0-CNF1/ 12.0
      GX(6)= 1.0- 4.0/CNF1
      GX(7)= 1.0- CNF2/ 12.0
      GX(8)= 1.0- 4.0/ CNF2
      GX(9)= 1.0- SIAD1/ 35.0
      GX(10)= 1.0- 15.0/ SIAD1
      GX(11)= 1.0- SIAD2/ 35.0
      GX(12)= 1.0- 15.0/ SIAD2
      GX(13)= 1.0- WD1/ WW1
      GX(14)= 1.0- WD2/ WW2
      GX(15)= 1.0- HP/ 8PW1
      GX(16)= 1.0- HP/ HPW2
      GX(17)= 1.0- H7/ HPB1
      GX(18)= 1.0- HP/ HPB2
      GX(19)= 1.0- (SC1*CT1*CR1)/ SALB1*CL1*CH1
      GX(2n)= 1.0- (SC2*CT2*CR2)/ SALB2*CL2*CH2
      GX(21)= 1.0- ERR/ 0.0012
      GX(22)= 1.0- 0.0005/ ERR
      GX(23)= 1.0- 0.6*DG2/ CD3
      GX(24)= 1.0- ABS(0.5*CD3- CD1*COS(BETA))/ ( 0.25* CD3)
      GX(25)= 1.0- B1/( 6.0* PND1)
      GX(26)= 1.0- (1.5*PND1)/ B1
      GX(27)= 1.0- B2/( 6.0*PND2)
      GX(28)= 1.0- (1.5*PND2)/ B2
      GX(29)= 1.0- (2.0*DSH1)/ DP1
      GX(30)= 1.0- (2.0*DSH2)/ DP2
      GX(31)= 1.0- (2.0*DSH3)/ DG2
      GX(32)= 1.0- DP2/(2.0 (CD1- 15.0))
      GX(33)= 1.0- ALPHD/90.0
      GX(34)= 1.0- 30.0/ ALPHD
      GX(35)= 1.0- TRAN1/ 7.0
      GX(36)= 1.0- TRAN2/ 7.0
      XX= GX(1)
      DO 8 I= 1, NC
      IF( XX.GE.GX(I)) XX= GX(I)
    8 CONTINUE
      IF( XX.LT.0.0) GO TO 9
      SUM= n.0
      DO 10 I= 1, NC
   10 SUM= SUM+R/ GX(I)
      FP= SUM
      FO= FO**SENS
      FT= FO+FP
      GO TO 11
```

```
 9     FT= 10.0E+30
11     DO 5 I= 1, N
5      X(I)= X1(I)
       RETURN
       END
```

```
$IBFTC UNCONS
      SUBROUTINE UNCONS(X,N,R)
      DIMENSION  X(15), D(15), GX(40), Y(15)
      DIMENSION G(15), G1(15), H(15, 15), GX1(40), GX2(40)
      COMMON/ DMS 10/ EPS1, EPS2, EPS3, EPS4, ITIR
       COMMON/ DMS 15/ GX, GX1, GX2
       COMMON/ DMS 17/ D, Y, G
 15    FORMAT(  // 10X,   *COUNTER NOT SUFFICIENT FOR MINIMISATION*)
 16    FORMAT( // 10X,  *NORMAL CONVERGENCE AFTER*,I3,*  ITIRATION*)
 25    FORMAT( / 10X,* INITIAL DESIGN VECTOR *, 5F6.3 )
 30    FORMAT(// 10X,*MINIMISATION OVER R *,F15.6,3X,*STARTS*//)
 41    FORMAT(// 10X, * LESS THAN THE LEAST VALUE OF X *)
 42    FORMAT( /  10X, * GRADIENT =*,6E15.6 )
 43    FORMAT( /  10X,* DIRECTION =* 6E15.6 )
 44    FORMAT(    10X,* STEP-SIZE =*, E15.6 )
 49    FORMAT( / 10X,* FROM UNCONS *, 5X, * FO =*,E15.6,* FP =*,E15.6,
     1 * FT1 =*, E15.6 )
 50    FORMAT(     10X, * ANGLE =*, F6.3 )
 51    FORMAT(/ 10X, * MODIFYING MATRIX * )
 132  FORMAT (15X,*DERIVATIVE OF FUNCTION ALONG DIRECTION OF *,/,15X,
     1*LINEAR MINIMISATION IS NOT NEGATIVL AT*,I4,*  ITIRATION FOR R=*,0
     2F10.6)
       PRINT 30 ,R
       PRINT 25,(X(I),I=1,N)
       L7 = 0
       L1 = 0
       CALL PENAL(FO,FP,FT,X,N,R,GX)
       IF( FT. GT. 10.0E+20 ) GO TO 17
       CALL GRAD( X,N,G,R)
       PRINT 42,(G(I),I=1,N)
 1     DO 5 I3 = 1,N
       DO 5 I2 = 1,N
       H(I2,I3) = 0.0
 5     H(I3,I3) = 1.0
 8     DO 6 I1 = 1,N
       D(I1) = 0.0
       DO 6 I2 = 1,N
 6     D(I1) = D(I1)-H(I1,I2)*G(I2)
       S9 = 0.0
       S8 = 0.0
       DO 10 I = 1,N
        S8= S8+D(I)*D(I)
 10    S9 = S9+D(I)*G(I)
```

```
      IF(S9.LT.0.0) GO TO 2
      DO 23 I2 = 1,N
      IF(H(I2,I2).NE.1.0)GO TO 1
23    CONTINUE
      PRINT 132,L1,R
      RETURN
2     XP = 1.0
      DO 199 I = 1,N
      IF(X(I).LT.XP) XP = X(I)
199    D(I) = D(I)/SQRT(S8)
       PRINT 43, ( D(I), I= 1, N)
      XP = XP*0.001
      CALL SEARCH(SIG,X,N,R)
       IF( SIG. LE. 10.0E-6 ) GO TO 13
      SIG = SIG/SQRT(S8)
      PRINT 44,SIG
      IF (SIG.LT.XP) L7 = L7+1
       IF( L7. GT. 10 ) GO TO 40
      DO 7 I1= 1,N
      D(I1) = D(I1)*SQRT(S8)
      X(I1) = X(I1)+SIG*D(I1)
7     G1(I1) = G(I1)
      PRINT 25,(X(I),I=1,N)
      L1 = L1+1
      IF (L1.GT.100) GO TO 14
      CALL PENAL (FO,FP,FT1,X,N,R,GX)
      PRINT 49,FO,FP,FT1
      SS = ABS((FT1-FT)/FT)
      FT = FT1
      CALL GRAD(X,N,G,R)
      PRINT 42,(G(I),I=1,N)
      ET = AMAX1(1.0,SIG)
      L2 = 0
      DO 3 I = 1,N
      IF(ABS(ET*D(I)).LT.0.001) GO TO 3
      L2 = L2+1
3     CONTINUE
      IF(L2.EQ.0) GO TO 13
      S1 = 0.0
      S2 = 0.0
      S3 = 0.0
      DO 11 I = 1,N
      S1 = S1+G(I)**2
      S2 = S2+D(I)**2
11    S3 = S3+G(I)*D(I)
      S2 = SQRT(S2)
      S1 = SQRT(S1)
      SG = ABS(S3/S2/S1)
      PRINT 50,SG
      IF (ABS(S3/S1/S2).GT.0.1) GO TO 8
      CXCX = 1.0E-2
      IF (S1.LE.CXCX.OR.S2.LE.CXCX) GO TO 13
```

```
 151 DO 9 I1 = 1,N
 9    Y(I1) = G(I1)-G1(I1)
      S11 = 0.0
      S12 = 0.0
      DO 152 I = 1,N
      S11 = S11+D(I)*Y(I)
 152  S12 = S12+Y(I)*G(I)
      SUM2 = 0.0
      DO 156 I = 1,N
      SUM1 = 0.0
      DO 157 J = 1,N
 157   SUM1 = SUM1+X(J)*H(J,I)
 156    SUM2 = SUM2+X(I)*SUM1
      A = SUM2
      IF(A.LT.0.0) GO TO 1
      PRINT 51
      CALL HMATRI(H,SIG,N)
      GO TO 8
 14   PRINT 15
      GO TO 17
 40   PRINT 41
      GO TO 17
 13   PRINT 16, L1
 17   RETURN
      END


$IBFTC MODIF
      SUBROUTINE MODIF( XO, N, NC, R )
       DIMENSION XO(15), XN(15), SPAN(15), X(15)
       DIMENSION PNDA(30),  PNDB(30), XMIN(15)
       DIMENSION  GX(40), GX1(40), GX2(40)
       COMMON/DMS5/DP1,DP2,DG1,DG2,B1,B2,ZG1,ZG2,PLV1,PLV2,COSB
       COMMON/ DMS 8/ CD1, CD2, CD3,OZP1,OZG1, OZP2, OZG2
       COMMON/DMS11/ ZP1, ZP2, PND1, PND2, SIAD1, SIAD2
       COMMON/ DMS12/ SI11, SIA2, TRAN1, TRAN2, ALPHD, ALPHA
       COMMON/ DMS 13/ CNF1, CNF2, ERR, DSH1, DSH2, DSH3, SC1, SC2
       COMMON/ DMS 14/ XMIN, XN, SPAN
       COMMON/ DMS 15/ GX, GX1, GX2
       COMMON/DMS 19/ PMOD1, PMOD2, TRAN,      FOB
C      ******************************Q*********************************
C
C      THE FOLLOWING ARE THE OPTIMUM DESIGN VARIABLES WITHOUT ROUNDING
C      OFF THE NO OF TEETH AND WITHOUT USING THE PREFERRED MODULES
C
C      ****************************************************************
       DO 90 I= 1, N
 90    XO(I)= XO(I)* SPAN(I)+ XMIN(I)
       ZP1= XO(1)
       ZP2= XO(2)
```

```
      PND1= XO(3)
      PND2= XO(4)
      SIAD1= XO(5)
      SIAD2= XO(6)
      TRAN1= XO(7)
      ALPHD= XO(8)
      CNF1= XO(9)
      CNF2= XO(10)
      ERR= XO(11)
      PMOD1= 25.4/ PND1
      PMOD2= 25.4/ PND2
      TRAN2= 20.0/ TRAN1
      ZG1= ZP1* TRAN1
      ZG2= ZP2* TRAN2
      OZP1= ZP1
      OZP2= ZP2
      OZG1= ZG1
      OZG2= ZG2
      READ 10, ( PNDA(I), I= 1, 30 )
      READ 10, ( PNDB(I), I= 1, 30 )
 10   FORMAT( 8 F10.5 )
C     ****************************************************************
C
C     ROUNDING OF THE NUMBER OF TEETH OF PINION AND GEAR
C
C     ****************************************************************
      IZP1= ZP1
      AZP1= IZP1
      IF( ( ZP1-AZP1). GE. 0.5) AZP1= AZP1+ 1.0
      IZP2= ZP2
      AZP2= IZP2
      IF(( ZP2-AZP2). GE. 0.5 ) AZP2= AZP2+ 1.0
      IZG1= ZG1
      AZG1= IZG1
      IF(( ZG1-AZG1). GE. 0.5 ) AZG1= AZG1+ 1.0
      IZG2= ZG2
      AZG2= IZG2
      IF(( ZG2-AZG2). GE. 0.5 ) AZG2= AZG2+ 1.0
C     ****************************************************************
C
C     RE- STORING THE ROUNDED OFF VARIABLES IN THE ORIGINAL NAME
C
C     ****************************************************************
      ZP1= AZP1
      ZP2= AZP2
      ZG1= AZG1
      ZG2= AZG2
      TRAN1= ZG1/ ZP1
      TRAN2= ZG2/ ZP2
      TRAN= TRAN1* TRAN2
```

```
      XO(1)= ZP1
      XO(2)= ZP2
      XO(7)= TRAN1
      DO 15 I= 1, 30
      J= I+ 1
      I1= I
      J1= I1+ 1
      IF(PMOD1. LE. PNDA(J). AND. PMOD1. GE. PNDA(I)) GO TO 16
 15   CONTINUE
 16   PMOD1= PNDA(J)
      DO 17 I= 1, 30
      J= I+ 1
      I2= I
      J2= I2+ 1
      IF( PMOD2. LE. PNDB(J). AND. PMOD2. GE. PNDB(I)) GO TO 18
 17   CONTINUE
 18   PMOD2= PNDB(J)
      PND1= 25.4/ PMOD1
      PND2= 25.4/ PMOD2
      XO(3)= PND1
      XO(4)= PND2
      DO 20 L= 1, N
      XO(L)= ( XO(L)-XMIN(L))/ SPAN(L)
 20   X(L)= XO(L)
      CALL PENAL( FO, FP, FT, X, N, R, GX )
      DO 24 K= 1, NC
 24   IF( GX(K). LT.0.0. OR. GX(K). GT. 1.0 ) GO TO 30
      GO TO 25
 30   PMOD1= PNDA(I1 )
      PMOD2= PNDB(I2 )
      X(3)= 25.4/ PMOD1
      X(4)= 25.4/ PMOD2
      X(3)= ( X(3)- XMIN(3))/ SPAN(3)
      X(4)= ( X(4)- XMIN(4))/ SPAN(4)
      CALL PENAL( FO, FP, FT, X, N, R, GX )
      DO 26 K= 1, NC
 26   IF( GX(K). LT. 0.0. OR. GX(K). GT. 1.0 ) GO TO 35
      GO TO 25
 35   PMOD1= PNDA(I1)
      PMOD2= PNDB(J2)
      X(3)= 25.4/ PMOD1
      X(4)= 25.4/ PMOD2
      X(3)= ( X(3)- XMIN(3))/ SPAN(3)
      X(4)= ( X(4)- XMIN(4))/ SPAN(4)
      CALL PENAL( FO, FP, FT, X, N, R, GX )
      DO 27 K= 1, NC
 27   IF( GX(K). LT. 0.0. OR. GX(K). GT. 1.0 ) GO TO 40
      GO TO 25
 40   PMOD1= PNDA(J1)
```

```
      PMOD2= PNDB(I2)
      X(3)= 25.4/ PMOD1
      X(4)= 25.4/ PMOD2
      X(3)= ( X(3)- XMIN(3))/ SPAN(3)
      X(4)= ( X(4)- XMIN(4))/ SPAN(4)
      CALL PENAL( FO, FP, FT, X, N, R, GX )
      DO 28 K= 1, NC
 28   IF( GX(K). LT. 0.0. OR. GX(K). GT. 1.0 ) GO TO 55
      GO TO 25
 25   CONTINUE
      GO TO 75
 55   I1= J1
      J1= J1+ 1
      I2= J2
      J2= J2+ 1
      IF( J1. GE. 28. OR. J2. GE. 28 ) GO TO 76
      GO TO 30
 76   CONTINUE
      IZG2= ZG2
      AZG2= IZG2
      IF((ZG2-AZF2). GE. 0.5) AZG2= AZG2+1.0
      ZG2= AZG2
 75   DO 60 I= 1, N
 60   XO(I)= X(I)
      DP1= ZP1* PMOD1
      DP2= ZP2* PMOD2
      DG1= ZG1* PMOD1
      DG2= ZG2*PMOD2
      RETURN
      END


$IBFTC DESIGN
      SUBROUTINE DESIGN( XO, N, NC )
      DIMENSION  XO(15), XMIN(15), SPAN(15), X(15), XN(15)
      COMMON/DMS5/DP1,DP2,DG1,DG2,B1,B2,ZG1,ZG2,PLV1,PLV2,COSB
      COMMON/ DMS 8/ CD1, CD2, CD3,OZP1,OZG1, OZP2, OZG2
      COMMON/ DMS11/ ZP1, Z72, PND1, PND2, SIAD1, SIAD2
      COMMON/ DMS12/ SIA1, SIA2, TRAN1, TRAN2, ALPHD, ALPHA
      COMMON/ DMS 13/ CNF1, CNF2, ERR, DSH1, DSH2, DSH3, SC1, SC2
      COMMON/ DMS 14/ XMIN, XN, SPAN
      COMMON/ DMS16/ HPW1, HPW2, HPB1, HPB2, HEIGHT, BASE, AREA
      COMMON/DMS18/ CORRP1, CORRG1, CORRP2, CORRG2
      COMMON/DMS 19/ PMOD1, PMOD2, TRAN,       FOB
      DO 90 I= 1, N
 90   XO(I)= XO(I)* SPAN(I)+ XMIN(I)
      ZP1= XO(1)
      ZP2= XO(2)
      PND1= XO(3)
      PND2= XO(4)
```

```
      SIAD1= XO(5)
      SIAD2= XO(6)
      TRAN1= XO(7)
      ALPHD= XO(8)
      CNF1= XO(9)
      CNF2= XO(10 )
      ERR= XO(11)
      ALPH1= ALPHD*3.1416/ 180.0
      ALFA= 20.0* 3.1416/ 180.0
C     ******************************************************************
C
C     ROUNDING OFF THE CENTRE DISTANCES
C
C     ******************************************************************
      CEN1= CD1
      CEN2= CD2
      ICEN1= CEN1
      ACEN1= ICEN1
      IF((CEN1-ACEN1). GE. 0.5 ) ACEN1= ACEN1+ 1.0
      ICEN2= CEN2
      ACEN2= ICEN2
      IF((CEN2-ACEN2). GE. 0.5 ) ACEN2= ACEN2+ 1.0
      CEN1= ACEN1
      CEN2= ACEN2
      CD1= CEN1
      CD2= CEN2
      CD33= (CD1**2)+(CD2**2)-2.0*CD1*CD2*COS(ALPHA)
      CD3=  SQRT(CD33)
      COSB= ((CD1**2)+(CD3**2)-(CD2**2))/(2.0*CD1*CD3)
      BETA= ARCOS(COSB)
      BASE= CD3
      HEIGHT= CD1* SIN(BETA)
      AREA= CD3*CD1*SIN(BETA)/(2.0*144.0 )
      FOB= AREA
C     ******************************************************************
C
C     ESTIMATION OF TOTAL CORRECTION, XT= 2X
C
C     ******************************************************************
      Q1= (OZP1+ OZG1)*COS(ALFA)/(ZP1+ZG1)
      A31= ARCOS(Q1)
 15   FORMAT( / 10X, 2 E15.4 )
      PRINT 15, Q1, A31
      XT1= (TAN(A31)-A31-TAN(ALFA)+ALFA)*(ZP1+ZG1)/(2.0*TAN(ALFA))
      Q2= (OZP2+OZG2)*COS(ALFA)/(ZP2+ZG2)
      A32= ARCOS(Q2)
      PRINT 15, Q2, A32
      XT2= (TAN(A32)-A32-TAN(ALFA)+ALFA)*(ZP2+ZG2)/(2.0*TAN(ALFA))
```

```
      CALL CORREC( ZP1, ZG1, TRAN1, XT1, XP, XG )
      CORRP1= XP
      CORRG1= XG
      CALL CORREC( ZP2, ZG2, TRAN2, XT2, XP, XG )
      CORRP2= XP
      CORRG2= XG
      V1= PLV1/3.28
      V2= PLV2/3.28
      X1= CORRP1
      X2= CORRG1
      X3= CORRP2
      X4= CORRG2
      CORRP1= X1
      CORRG1= X2
      CORRP2= X3
      CORRG2= X4
      IQLTY= ERR
      IQLTY= 7
C     ******************************************************************
C     CALCULATION OF MEASUREMENT DATA
C     ******************************************************************
      DP1=25.4* DP1
      DP2= 25.4*DP2
      DG1= 25.4* DG1
      DG2= 25.4* DG2
      B1= B1* 25.4
      B2= B2* 25.4
      PLV1= PLV1/(3.28*60.0)
      PLV2= PLV2/(3.28*60.0)
      ERR= ERR* 25.4
      BASE= BASE* 25.4
      HEIGHT= HEIGHT* 25.4
      AREA= AREA/(3.28*3.28 )
      RETURN
      END



$IBFTC CORREC
      SUBROUTINE CORREC( ZP, ZG, TRNS, XT, XP, XG )
      DIMENSION  XM(12), SLP(12), XM1(13), SLP1(13)
      COMMON/ DMS16/ HPW1, HPW2, HPB1, HPB2, HEIGHT, BASE, AREA
      DATA XM/-0.35, -0.25, -0.14, -0.05, 0.08, 0.17, 0.26, 0.38= 0.50,
     1 0.62, 0.72, 0.84 /
      DATA SLP/-0.8935, -0.7535, -0.6085, -0.4835, -0.3775, -0.2735,
     1 -0.1762, -0.0906,-0.00436, +0.0567, +0.01351, +0.1942 /
      DATA XM1/ -0.38, -0.18, -0.12, -0.06, 0.00, 0.07, 0.15, 0.23,
     1 0.34, 0.44, 0.57, 0.72, 0.9 /
      DATA SLP1/ -0.3060, -0.0559, -0.0426, -0.0174, 0.0, 0.00332,
     10.0663, 0.1050, 0.1620, 0.2325, 0.3205, 0.4750, 0.6850 /
```

```
      X=XT/ 2.0
C
C
C
C
C     DISTRIBUTION OF CORRECTION
      ZZ= ( ZP + ZG)/ 2.0
      IF( TRNS. GT. 1.0 ) GO TO 731
C     FOR SPEEDING DOWN PAIR OF GEARS
      XX1= SLP(1)*(ZZ-40.0)/ 50.0+ XM(1)- X
      DO 735 N= 2, 12
      XX2= SLP(N)*( ZZ-40.0)/ 50.0+ XM(N)- X
      IF(XX1*XX2 ) 736, 736, 734
 734  XX1= XX2
 735  CONTINUE
 736  SL= SLP(N-1)-XX1*( SLP(N)-SLP(N-1))/(XX2-XX1)
      GO TO 751
C     FOR SPEEDING UP PAIR OF GEARS
 731  IF(X.GE.(-0.3)) GO TO 732
      IF(ZZ. GT. 50.0 ) GO TO 739
 732  XX1= SLP1(1)*( ZZ-40.0)/ 50.0+ XM1(1)- X
      DO 795 N= 2, 13
      XX2= SLP1(N)*(ZZ-40.0)/ 50.0+ XM1(N)- X
      IF(XX1* XX2) 737, 737, 738
 738  XX1= XX2
 795  CONTINUE
 737  SL= SLP1(N-1)-XX1*(SLP1(N)- SLP1(N-1))/ (XX2-XX1 )
 751  XP= SL*(ZP-ZZ)/ 50.0+ X
      XG= XT- XP
      GO TO 761
 739  XP= X
      XG= X
 761  RETURN
      END


$IBFTC GRAD
      SUBROUTINE GRAD( X, N, G, R)
      DIMENSION  X(15), GX1(40), GX2(40), G(15)
      DIMENSION  GX(40)
       COMMON/ DMS1/ PI, RAD, NC
       COMMON/ DMS 15/ GX, GX1, GX2
      CALL PENAL ( FO, FP, FT, X, N, R, GX1 )
      DO 2 I= 1, N
      A= X(I)
      ST= AMIN1( 0.001, A/100.0 )
      X(I)= X(I)+ ST
      CALL PENAL ( FO1, FP, FT, X, N, R, GX2 )
      G(I)= ( FO1-FO)/ ST
      DO 3 J= 1, NC
```

```
      G(I)= G(I)- (( GX2(J)- GX1(J))/ST)/GX1(J)**2 *R
 3    CONTINUE
      X(I)= X(I)- ST
 2    CONTINUE
      RETURN
      END
```

```
$IBFTC HMAT
      SUBROUTINE HMATRI ( H, SIG, N )
       COMMON/ DMS 17/ D, Y, G
      DIMENSION  H5(15), H9(15, 15), H10(15, 15),  HS(15)
      DIMENSION  H6(15,15), H(15,15), Y(15), D(15), H4(15)
      DIMENSION  G(15)
C
       H1= 0.0
      DO 1 I= 1, N
      H1= H1+ D(I)* Y(I)
      H5(I)= 0.0
      H4(I)= 0.0
      DO 1 J= 1, N
       H4(I)= H4(ID+H(I,J)*Y(J)
      H5(I)= H5(I)+ H(J,I)* Y(J)
 1    H9(I, J)=  D(I)* D(J)
      H11= 0.0
      DO 2 I= 1, N
      H11= H11+ Y(I)* H4(I)
      DO 2 J= 1, N
 2    H10(I,J)= H4(I)* H5(J)
      DO 3 I= 1, N
      DO 3 J= 1, N
      H6(I, J)= H(I,J)+ SIG/ H1*H9(I,J)- H10(I,J)/ H11
 3    CONTINUE
      DO 5 I= 1, N
      DO 5 J= 1, N
 5    H( I, J)= H6(I, J)
      RETURN
      END
$ENTRY
```

SAMPLE OUTPUT FOR EXAMPLE-2

INPUT DATA

VALUES FOR CO1, CR1, CL1, CS1, CH1, CT1, CF1 FACTORS

1.5000 1.2500 1.1000 1.0000 1.0000 1.0000 1.0000

VALUES FOR CO2, CR2, CL2, CS2, CH2, CT2, CF2 FACTORS

1.5000 1.2500 1.1000 1.0000 1.0000 1.0000 1.0000

VALUES FOR CF, SENS. FACTORS

10.0000 1.0050

VALUES FOR THE MINIMUM VALUES FOR THE DESIGN VARIABLES

13.7000 13.7000 0.5000 0.5000 15.0000 15.0000 2.0000

30.0000 5.0000 5.0000 0.0005

VALUES FOR THE DESIGN VARIABLES

100.0000 130.0000 2.0000 2.0000 28.0000 28.0000

5.0000 70.0000 8.0000 8.0000 0.0008

VALUES FOR THE SPAN OF THE DESIGN VARIABLES

250.0000 250.0000 15.0000 15.0000 20.0000 20.0000

5.0000 60.0000 10.0000 10.0000 0.0007

VALUES FOR EPS1, EPS2, EPS3, EPS4

0.0005 0.0005 0.0005 0.0005

NO OF ITERATIONS = 200

OBJECT FUNCTION 242.714

PENALTY FUNCTION 81.547

TOTAL FUNCTION 344.492

CONSTRAINTS

0.86300 0.89462 0.88680 0.89037 0.33333 0.50000 0.33333 0.50000
0.20000 0.46429 0.20000 0.46429 0.90473 0.77785 1.00000 0.91678
0.96397 0.52811 1.00000 0.41812 0.29167 0.41176 0.07274 0.72398
0.09310 0.72434 0.09310 0.72434 0.84827 0.80053 0.92087 0.76279
0.22222 0.57143 0.28571 0.42857

INITIAL VALUE OF R = 200.00

MINIMISATION OVER R = 2.00.00000 STARTS

NORMAL CONVERGENCE AFTER 25 ITERATION

OBJECT FUNCTION 590.172

PENALTY FUNCTION 14305.574

TOTAL FUNCTION 14895.746

MINIMISATION OVER R = 20.00000 STARTS

NORMAL CONVERGENCE AFTER 23 ITERATION

OBJECT FUNCTION 251.449

PENALTY FUNCTION 1533.859

TOTAL FUNCTION 1785.308

MINIMISATION OVER R = 2.00000 STARTS

LESS THAN THE LEAST VALUE OF X

OBJECT FUNCTION 150.152

PENALTY FUNCTION 187.348

TOTAL FUNCTION 337.500

MINIMISATION OVER R = 0.20000 STARTS

NORMAL CONVERGENCE AFTER 20 ITERATION

OBJECT FUNCTION 109.515

PENALTY FUNCTION 31.404

TOTAL FUNCTION 140.919

MINIMISATION OVER R = 0.020000 STARTS

NORMAL CONVERGENCE AFTER 15 ITERATION

OBJECT FUNCTION 98.861

PENALTY FUNCTION 6.405

TOTAL FUNCTION 105.265

MINIMISATION OVER R = 0.002000 STARTS

LESS THAN THE LEAST VALUE OF X

OBJECT FUNCTION 95.882

PENALTY FUNCTION 1.572

TOTAL FUNCTION 97.453

MINIMISATION OVER R = 0.000200 STARTS

LESS THAN THE LEAST VALUE OF X

OBJECT FUNCTION 95.640

PENALTY FUNCTION 0.218

TOTAL FUNCTION 95.859

NORMALISED OPTIMUM DESIGN VARIABLES

0.087061	0.254675	0.123583	0.122399	0.399731
0.399727	0.801157	0.373471	0.375708	0.687293
0.012267				

ACTUAL DESIGN VARIABLES

35.4652	77.3688	2.3537	2.3360	22.9946
22.9945	6.0058	52.4083	11.7571	11.8720
0.0005				

INPUT DESIGN DATA

HORSE POWER = 60000.000

TURBINE SPEED = 3000.0RPM

PROPELLER SPEED = 150.000RPM

OVERALL REDUCTION RATIO = 20.0

OPTIMUM GEAR DESIGN

	FIRST PINION	FIRST GEAR	SECOND PINION	SECOND GEAR
NUMBER OF TEETH	34	212	76	254
P C D IN M M.	442.00	2756.00	912.00	3048.00
FACE WIDTH IN MM.	415.84	415.84	387.63	387.63
MODULES IN M.M.	13.00	13.00	12.00	12.00
CORREC FACTOR	-0.073	-.158	-.42	-.42
HELIX ANGLE IN DEGREE	22.99	22.99	22.99	22.99
PROFILE CONTACT RATIO	0.69	0.69	0.73	0.73
FACE CONTACT RATIO	5.09	5.09	5.14	5.14
P L VELOCITY IN M P S	82.55	82.55	27.55	27.55
HORSE POWER IN WEAR	128208.55	128208.55	320922.92	320922.92
HORSE POWER IN BENDING	1521862.27	1521862.27	93157.83	93157.83
TRANSMISSION RATIO	6.09	6.09	3.29	3.29
ALLOW MANF ERR IN M M.	0.0129	0.0129	0.0129	0.0129

GEOMETRY OF GEAR SET

BASE = 1868.79 MM

HEIGHT= 1797.23 MM

AREA = 1.68 SQ.METS